ARI Research Note 99-26

Research Into the Use of Speech Recognition Enhanced Microworlds in an Authorable Language Tutor

Beth Plott, Andrew Hamilton, and Eric Princen
Micro Analysis and Design, Inc.

Stephen LaRocca and John Morgan
U.S. Military Academy

Jonathan D. Kaplan
U.S. Army Research Institute

Research and Advanced Concepts Office
Michael Drillings, Chief

April 1999

19990427 027

U.S. Army Research Institute
for the Behavioral and Social Sciences

Approved for public release; distribution is unlimited.

U.S. Army Research Institute for the Behavioral and Social Sciences

A Directorate of the U.S. Total Army Personnel Command

EDGAR M. JOHNSON
Director

Research accomplished under contract
for the Department of the Army

Micro Analysis and Design, Inc.

Technical review by

Tony Papa

NOTICES

REPORT DOCUMENTATION PAGE

1. REPORT DATE (dd-mm-yy)	2. REPORT TYPE	3. DATES COVERED (from. . . to)
April 1999	Final	March 1997 – January 1999

4. TITLE AND SUBTITLE Research Into the Use of Speech Recognition Enhanced Microworlds in an Authorable Language Tutor	5a. CONTRACT OR GRANT NUMBER DASW01-97-C-0039 5b. PROGRAM ELEMENT NUMBER 622785
6. AUTHOR(S) Beth Plott, Andrew Hamilton, Eric Princen (Micro Analysis and Design, Inc.), Stephen LaRocca, John Morgan (U.S. Military Academy), Jonathan D. Kaplan (U.S. Army Research Institute)	5c. PROJECT NUMBER A790 5d. TASK NUMBER 2233 5e. WORK UNIT NUMBER C2
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Micro Analysis and Design, Inc. 4900 Pearl E. Circle, #201E Boulder, CO 80301	8. PERFORMING ORGANIZATION REPORT NUMBER MA&D 6455
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) U.S. Army Research Institute for the Behavioral and Social Sciences 5001 Eisenhower Avenue Alexandria, VA 22333-5600	10. MONITOR ACRONYM ARI 11. MONITOR REPORT NUMBER Research Note 99-26

12. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT *(Maximum 200 words)*:

An earlier ARI sponsored MILT project was designed to investigate the possibility of using natural language processing (NLP) software to identify semantic and syntactic errors and provide the basis for state of the art dialogue exercises. One of the thirteen exercise types developed was the microworld exercise. A microworld is a software environment in which students can issue commands that are executed by animation routines in a game like atmosphere.

Once the first microworld exercise was completed and integrated into MILT, ARI funded the investigation of the use of discreet speech recognition technology in language learning using the microworld exercise as a basis.

The goal of this current effort was to expand the capabilities of MILT and incorporate continuous speech recognition for Arabic, Spanish and English. The overall objective of this project was to develop a general purpose, authorable, microworld that utilizes continuous speech recognition. The central tasks were 1) the design of an enhanced microworld exercise, 2) development of continuous speech recognition components for English, Arabic, and Spanish, 3) incorporation of speech recognition into the microworld exercise, and 4) expansion of the Arabic natural language processing (NLP) system.

15. SUBJECT TERMS

Continuous Speech Recognition · 3D Microworlds · Language Tutor · Authoring Tool

SECURITY CLASSIFICATION OF			19. LIMITATION OF ABSTRACT	20. NUMBER OF PAGES	21. RESPONSIBLE PERSON (Name and Telephone Number)
16. REPORT	17. ABSTRACT	18. THIS PAGE			
Unclassified	Unclassified	Unclassified	Unlimited	78	Dr. Jonathan Kaplan (703) 617-8828

RESEARCH INTO THE USE OF SPEECH RECOGNITION ENHANCED MICROWORLDS IN AN AUTHORABLE LANGUAGE TUTOR

EXECUTIVE SUMMARY

Research Requirement:

Micro Analysis and Design, Inc. and the University of Maryland developed the Military Language Tutor (MILT) under an earlier contract (MDA903-92-C-0229). The U.S. Army Research Institute was the primary sponsor of the MILT project. MILT was also sponsored through ARI by the Office of Special Technology of DoD, and the Defense Advanced Research Projects Agency (DARPA). Initially the MILT project was designed to investigate the possibility of using natural language processing (NLP) software to identify semantic and syntactic errors and provide the basis for state of the art dialogue exercises.

A graphical flowchart approach to designing language lessons and authoring templates for thirteen different exercise types were developed for the MILT project. One of the exercise types developed was the microworld exercise. A microworld is a software environment in which students can issue commands that are executed by animation routines in a game like atmosphere.

Once the first microworld exercise was completed and integrated into MILT, ARI funded the investigation of the use of discreet speech recognition technology in language learning using the microworld exercise as a basis. In the speech recognition microworld, students can issue spoken commands in Arabic by selecting among three displayed alternatives. The sequence of commands can be changed by the author. However, new commands cannot be easily added to the exercise.

The goal of this current effort was to expand the capabilities of MILT and incorporate continuous speech recognition for Arabic, Spanish and English. The resulting tool includes continuous speech recognition technology and significant improvements to the microworld exercise.

Procedure:

The overall objective of this project was to develop a general purpose, authorable, microworld that utilizes continuous speech recognition. The central tasks were 1) the design of an enhanced microworld exercise, 2) development of continuous speech recognition components for English, Arabic, and Spanish, 3) incorporation of speech recognition into the microworld exercise, and 4) expansion of the Arabic natural language processing (NLP) system.

Findings:

The effort resulted in the development of an authorable, speech recognition enhanced three dimensional microworld exercise and improved 32 bit MILT application.

Utilization of Findings:

The effort provides authoring tool and tutoring system that can be used to teach mission relevant information and language skills though the use of a three-dimensional environment and continuous speech recognition.

RESEARCH INTO THE USE OF SPEECH RECOGNITION ENHANCED MICROWORLDS IN AN AUTHORABLE LANGUAGE TUTOR

CONTENTS

LIST OF TABLES

LIST OF FIGURES

Overview

This report documents the development of an authorable tutoring system. Specifically, the focus was on the development of an authorable, three-dimensional, speech recognition enhanced microworld exercise capable of supporting military and language informational needs.

As part of this effort the following major tasks were performed:

- an enhanced microworld exercise was designed and developed
- continuous speech recognition components for English, Arabic, and Spanish were developed
- speech recognition was incorporated into the 3D microworld exercise
- the Arabic natural language processing (NLP) system was expanded

The effort was based on expansion of the Army Research Institute sponsored MILT 1.0 project.

Results of the Work

In the contract, the overall objective was to develop a general purpose, authorable, enhanced microworld exercise that includes continuous speech recognition. The effort resulted in the successful completion of fourteen tasks.

- Task 1: Development of a Work Plan describing the steps to be taken and the projected time for completion
- Task 2: Acquisition of an appropriate continuous speech recognition system and any required expertise
- Task 3: Engage in on-site evaluation of discrete speech recognition version with SOF (Ft. Campbell, KY)
- Task 4: Develop a design for using speech recognition in the tutor
- Task 5: After review by ARI, personnel, revise the design
- Task 6: Implement the design
- Task 7: Develop Arabic CSR models
- Task 8: Develop Arabic, Spanish and English language models
- Task 9: Expand the Arabic NLP System
- Task 10: Deliver the software
- Task 11: Develop an Arabic continuous speech recognition driven microworld exercise
- Task 12: Prepare system documentation and user help
- Task 13: Complete monthly progress reports
- Task 14: Complete final report

These tasks are discussed in this section.

Task 1: Development of Work Plan

During this first task detailed estimates for each of the remaining tasks was made.

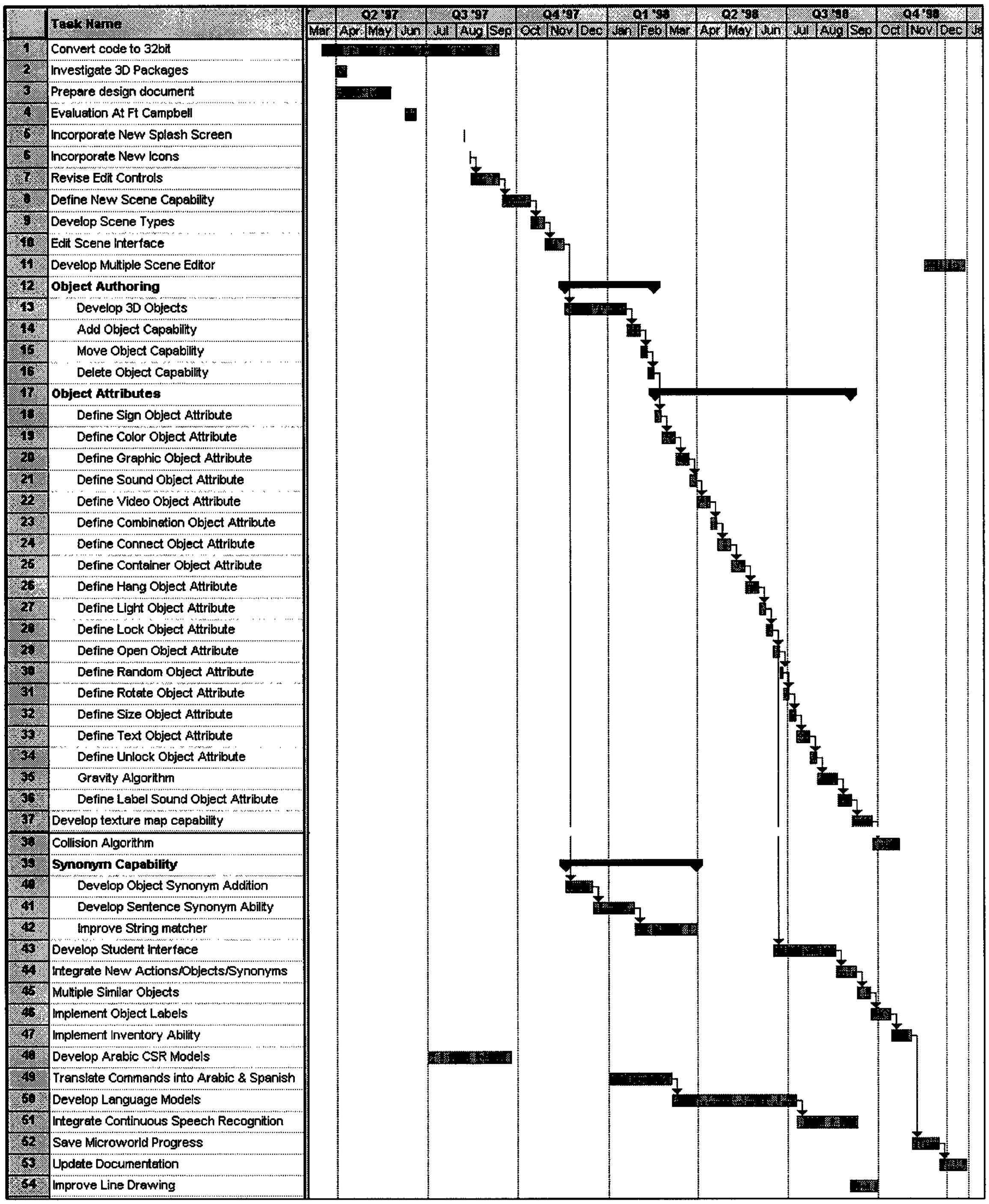


Figure 1 -Work Plan GANTT CHART

The estimate included time requirements, resource requirements, personnel requirements and critical subtask identification. A GANTT chart (Figure 1) was developed that shows each task in relation to the project schedule.

Task 2: Acquisition of an appropriate continuous speech recognition

Micro Analysis and Design evaluated several different CSR programs that were available in the first quarter of 1997. The commercial vendors considered were:

- Dragon Speech Systems
- Entropics Cambridge Research Laboratory
- Lernout & Hauspie
- Speech Systems
- IBM

It was decided that the Entropics HTK system would be the most desirable to integrate into MILT for the following reasons:

1. the accuracy of the CSR was rated highly
2. models had been built in languages other than English using the tool
3. it was fairly easy to develop language models in languages other than English
4. a development kit was available for incorporating the CSR into applications
5. other government agencies were using the system in products that it might be desirable to link to in the future
6. the software was executable on a PC Windows platform
7. the software was speaker independent
8. the licensing requirements were affordable
9. the CSR engine uses a standard PC sound card.

A Windows version of the Entropics HTK CSR software was purchased for use in this program.

The Army Research Institute and Micro Analysis and Design obtained CSR expertise from the U.S. Military Academy (USMA) for the translation of the Arabic utterances and development of the CSR acoustic and language models. The USMA has unique technical expertise in the development of Arabic Language Modules, including translation experience.

Task 3: Engage in on-site evaluation of discrete speech recognition version

Micro Analysis and Design and ARI prepared for and performed an on-site evaluation of the two-dimensional discrete speech interactive language tutor in MILT version 1.0. The evaluation was performed in June 1997 with Special Forces personnel at Ft. Campbell, KY.

Preparation for the evaluation involved:

- development of 2 Arabic speech interactive lessons (1 for lower level students and 1 for more advanced students)
- development of background questionnaire
- pre-test and post-test development
- development of microworld questionnaire
- development of survey questionnaire
- performance of sample evaluations by Arabic students in the DC area
- preparation of the equipment (computers, tape recorders, microphones, CDs, tapes, etc.)

At Ft. Campbell, the evaluation procedure followed was:

1. Briefly explain the purpose of the evaluation, including the unique functional capability of MILT as a tutoring system and the importance of his or her inputs, to the subject.
2. Subject completed the background questionnaire.
3. Conducted the pre-test: (a) gave the subject the microphone attached to the tape recorder and had him or her test its recording; (b) gave him or her an English version of the 70 utterances; (c) had him record his or her name, the date, and a word "pre-test"; (d) told him to speak each of the 70 English utterances in Arabic to record on the tape. Made sure that he or she was speaking each utterance in order; (e) after completing the recording, took out the audio tape and wrote the subject's name, date, and "pre-test" on the tape.
4. Told the subject to begin the listening and oral practice section. Started the subject on the MILT pronunciation exercise.
5. After completing the listening and oral practice section, gave the survey questionnaire prepared for that section.
6. As soon as the subject completed the questionnaire, told him or her to continue the next section of the tutor: oral production exercise in the micro-world.
7. When the subject completed the tutor, gave the questionnaire prepared for the micro-world section.
8. After completing the questionnaire, conducted the post-test: (a) gave the subject the microphone attached to the tape recorder and test its recording; (b) gave him or her the same English version of the 70 utterances used for the pre-test; (c) had him record his or her name, the date, and a word "post-test" on the tape; (d) told him to speak each of the 70 English utterances in Arabic to record on the tape. After completing the recording, took out the audio tape and wrote the subject's name, date, and "post-test" on the tape.
9. After completing the evaluation, the subject was thanked for his or her participation.

Twenty one subjects performed the evaluation.

The questionnaire and audio tapes were used to assess the usability of MILT from the subjective student assessments and from proficiency ratings of the Arabic speaking on the audio tapes.

The recorded audio tapes were delivered to a native Arabic speaking linguist with copies of the assessment form prepared for rating the proficiency level of Arabic usage.

The Arabic-speaking linguist:

(a) rated each of the four dimensions of Arabic (Vocabulary, Grammar, Pronunciation, Fluency) usage using a 5-point scale, "0" for "not being able to speak at all" and "5" for "excellent - perfect or near perfect";

(b) listened to at least a quarter of each tape before rating the subject's proficiency level on the assessment form;

(c) listened to the first utterance recorded for the pre-test and listen to the same utterance recorded for the post-test before rating them;

(d) listened to the first utterance recorded for the pre-test again and rated each of the four dimensions, and then listened to the same utterance recorded for the post-test again and rated it on each of the four dimensions; continued this process until all utterances were rated.

All four dimensions Vocabulary, Grammar, Pronunciation, and Fluency showed improvement in the post-test compared with the pre-test.

Task 4: Develop design

This task involved writing a design document that detailed the design for expanding the capabilities of MILT and adding continuous speech recognition. A document entitled "MILT 2.0 Design Document" was prepared and delivered to ARI in May 1997.

The "MILT 2.0 Design Document" described architectural changes (port to 32bit, and use of OpenGL 3D environment), significant microworld changes, the microworld authoring design, the design of the student perspective, and the designed use of speech recognition.

Task 5: After review by ARI, personnel, revise the design

ARI personnel reviewed the "MILT 2.0 Deign Document" and prepared a list of several comments and questions. MA&D personnel then revised the design document and submitted a revised document in June 1997.

Task 6: Implement the design

A significant amount of the effort on this contract was dedicated to implementing the design. This section describes the developed software.

32-bit Environment

MILT 1.x was a Windows 3.x compliant (16-bit) application. This allowed for its use on many common 16-bit systems. However, the 16-bit versions of windows lacked robustness and effective multitasking. The current family of 32-bit operating systems offered by Microsoft (Windows 95/98 and Windows NT 4.0) provide developers with a much more robust and 'user-friendly' operating environment. To take advantage of the 32-bit environment, the entire MILT application was reprogrammed using Microsoft Visual C++.

3D Graphic Environment

The microworld based lesson types in MILT 1.x were standard 2D graphics. With the expansion of MILT 2.0 into the 32-bit arena, 3D graphics APIs become available. The most industry tested API was Silicon Graphics' OpenGL. OpenGL has the ability to be accelerated by video cards that understand it to provide much faster rates of operation, but can also be handled completely by software.

There were other APIs such as Direct3D that were currently available for use in developing 3D graphics. MA&D conducted an investigation into which API would be serve the MILT purposes best. OpenGL was chosen as the API to use in development of MILT 2.0 due to the fact that it is widely supported by graphic application and hardware vendors and because other ARI programs were using OpenGL.

The MILT 3D Microworld was developed using an OpenGL development environment called Open Inventor. This software is "host ID locked." Host ID locking is a form of software license control that allows Open Inventor to run only on specifically licensed systems. Each time Open Inventor runs, it checks for a valid password.

Open Inventor uses both VRML (Virtual Reality Modeling Language) and Open Inventor objects. VRML is the file format adopted by the Internet community for 3D geometry data on the World Wide Web

To execute the MILT 3D software, users will need to obtain a runtime license for the host computer from Template Graphics Software (TGS). The web page for TGS is www.tgs.com.

Significant Microworld Changes

Microworld Overview

The microworld exercise in both versions 1.x and 2.0 is an authorable game-like exercise developed to complement the standard question types like multiple choice, question and answer. The authorability component of the microworld exercise makes it both complex and reusable. In traditional graphic games, once the user has gone through the game its utility has greatly diminished because the user already knows what objects are in each location and what the steps are to complete the game. The MILT microworld exercise can be used to develop virtually infinite numbers of different exercises. The author can change the question to be answered (the goal of the exercise), the location of objects, the attributes of objects, the background (texture) of each scene, the number of scenes available, and the answer feedback received (both incorrect and correct).

In the microworld exercise, the author develops a question for the student to answer. The student must find the answer by manipulating objects in one or more scenes. There are input modes for the microworld exercise: 1 mode requires the student to textually produce commands by typing on the keyboard, the second mode uses speech recognition technology to allow students to issue spoken commands to the MILT microworld.

This section points out the major changes that were made to the MILT microworld exercise for Version 2.0. Each of these changes is discussed in detail in the Authoring and/or Student Perspective sections.

Attributes and Objects-

Version 2.0 includes many more objects and attributes than Version 1.x. In addition, all of the low quality, bitmap based, 2D object graphics used in 1.x were completely redone using OpenGL to make much more realistic 3D graphic representations.

Actions

Actions are essentially verbs that the animated microworld "understands" and can perform. The action list for version 2.0 was significantly expanded from Version 1.x.

Labels

Students now have the ability to gain information about objects by looking at that object's "label". The label displays the name of the object and gives the student information about what actions can be performed on it (e.g. the book can be read, opened, closed, moved, etc.) Labels are displayed when the user selects an object with the mouse. The original screen text appears in English. Students are able to click a button to access the same information in the current foreign language. Authors can add synonyms that will be displayed to the student as object names. Authors cannot change the actions that can be performed on an object. Authors also have the option of recording one WAV sound file that the student can play from the Label screen.

First Person

The perspective in MILT 1.x is third person. It was decided to develop MILT 2.0 in first person so that the user can really feel like he/she is there.

Multiple Similar Objects

Through the use of complex algorithms and object attributes, MILT 2.0 allows more than one object of a certain type to be used in the same room. This also allows students to work with adjectives and allows students to carry objects from one scene to another.

Inventory

In version 1.x the agent could not carry items from scene to scene and could not just hold onto an item. For example, a legal command was "carry the book to the table", but "carry the book" was not. Commands such as "pick up" and "hold" were not recognized commands.

In version 2.0, MILT both the ability to carry objects from scene to scene and the ability to hold multiple items were developed. The ability to "inventory" multiple items made these new capabilities possible.

Saving Exercise Progress

In version 1.x a student could exit MILT during a lesson and had the option of saving their progress so that they could resume the lesson at a later time. MILT stored the student score thus far in the lesson and the exercise that should be resumed. However, MILT did not store individual microworld exercise status/progress – e.g. if a student exited MILT while in the 2nd room of a microworld exercise, when they resumed the lesson they would be placed at the beginning (1st room) of the microworld exercise.

It is envisioned that MILT 2.0 microworld exercises would be longer in duration with more scenes and objects than MILT 1.0. In order to support these longer exercises and allow the student to use the software over multiple sessions, the ability to save exercise progress was developed.

Improved Object Interactions

One of the complaints of MILT 1.0 was its lack of "physical" world knowledge. For example, there was no "gravity" or "collision" mechanism implemented. In MILT 2.0, the user is able to better manipulate objects and objects will interact with each other in a more natural fashion. For example, objects can collide with each other and objects will fall to the ground.

Object Synonym Authoring

MILT 1.0 was very limiting because the string matcher used did not allow synonyms for object names to be used by the student. The microworld only recognized one word to describe each object. This meant that if the student entered "Open the trashcan" instead of "Open the wastebasket" the command was not recognized. In MILT 2.0, authors can enter synonyms for objects.

Continuous Speech Recognition

In MILT 1.x, investigation into the use of discreet speech recognition in language learning was accomplished. The Dragon VoiceTools discreet recognition development system was used to create an Arabic user model and vocabulary file for 73 different microworld specific utterances and the speech recognition engine was integrated with MILT 1.0. Students issued spoken commands to the MILT microworld in Arabic by selecting among three alternative statements displayed at the bottom of the computer screen. The issuance of a command triggered the animation and caused three new commands to be displayed at the bottom of the screen. The author could change the sequence of commands. However, adding new commands required the use of the native dragon development software.

Several problems were identified with the MILT 1.x discreet recognition approach:

1. it resulted in oral reading practice, but not in natural speech production
2. since alternative commands were read from the screen, the natural language processing error analysis capability could not be used
3. the author was required to engage in a time-consuming effort to author the sequence of three possible commands displayed to the student

In Version 2.0, we attempted to solve these problems and expand the utility of MILT by incorporating a continuous speech recognition engine.

The continuous speech recognition software used is the HTK software. West Point Military Academy developed the Arabic language and acoustic continuous speech recognition modules for use in MILT 2.0. Micro Analysis and Design, Inc. developed the English and Spanish language modules. The English and Spanish Entropic HTK acoustic modules were used. The CSR development is discussed further in tasks 7 and 8.

Microworld Authoring Environment

Authoring Overview

The general authoring approach used in MILT to develop new lessons did not change in MILT 2.0. The MILT Authoring module was designed so that instructors without programming expertise can easily develop lessons. The lesson is defined using a graphical flowchart approach and each individual exercise is defined through the use of template screens. The MILT lessons are developed and stored in a hierarchical format. Authors create lessons, and lessons consist of one or more groups, or sets, of exercises.

Once the author has graphically defined an exercise, he can define the exercise type and parameters by clicking on an exercise node. MILT first displays the Exercise Attributes window (Figure 2).

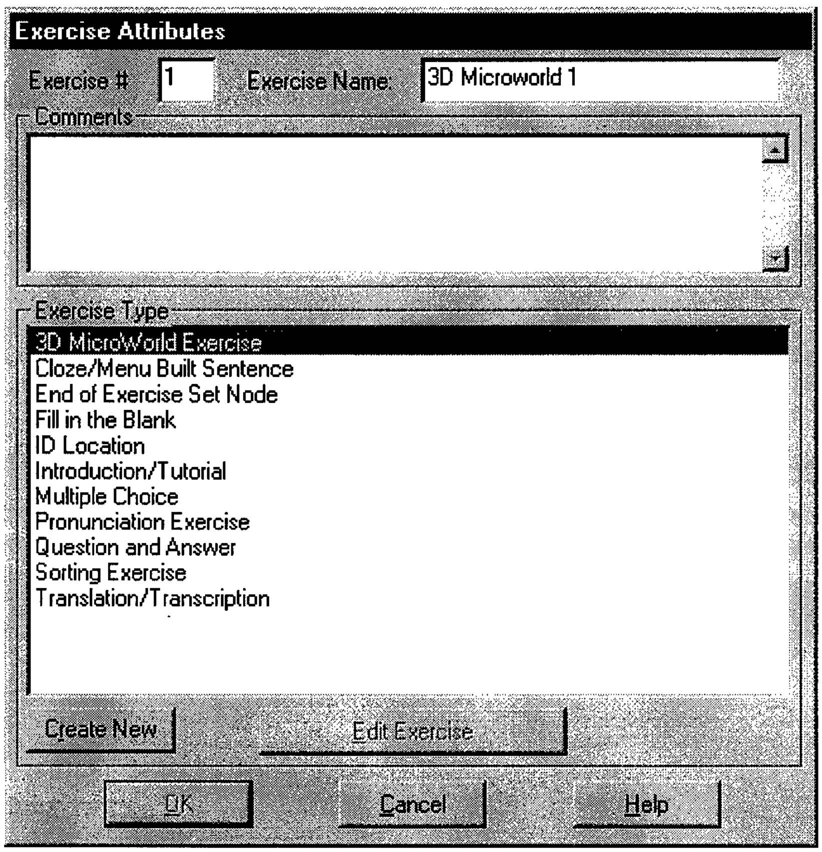


Figure 2 - Exercise Attributes Window

The Exercise Attributes window allows the author to define the type of exercise, name the exercise, renumber the exercise, record any comments about the exercise, and access the exercise parameters.

To begin defining a microworld exercise the author selects the '3D MicroWorld Exercise' option then clicks on the 'Edit Exercise' button.

Microworld Editor

After clicking 'Edit Exercise' MILT will display a window similar to Figure 3. From this window the author defines the exercise characteristics such as the number of scenes in the exercise, the exercise textures, the objects that will be included in each scene, the location of scenes, and object attributes.

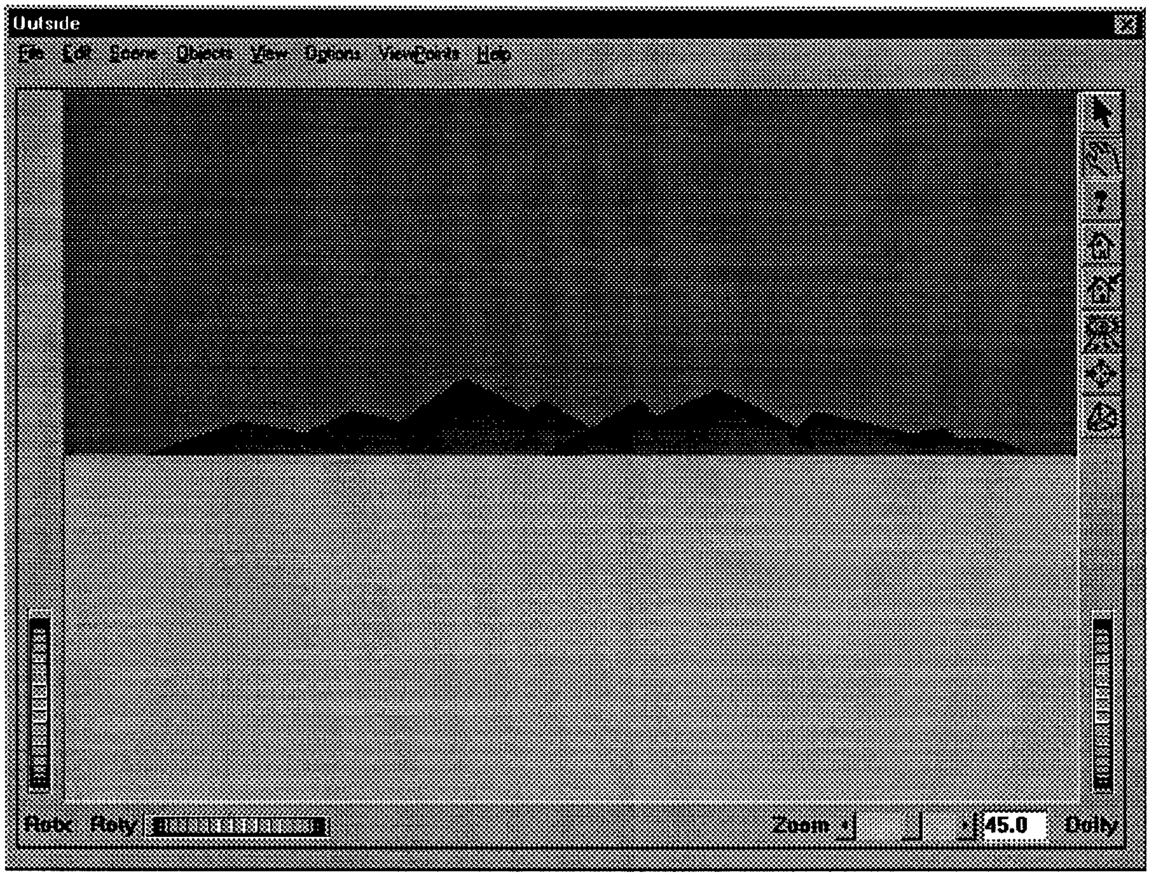


Figure 3 – Microworld Editor

The microworld exercise allows authors to link several rooms or scenes together in order to make a longer and more interesting game environment. A screen similar to Figure 3 shows the overall mapping that has been defined for the current microworld exercise. The author is shown an outside environment and places new scenes (buildings) and objects on this outside "microworld." The author can double click on any existing scene or select a scene name from the scene menu to begin editing any particular scene. The interface was designed so that when an author wants to change a given room that may be deep into the microworld scene structure, he can go there directly rather than going through all of the rooms that lead to it.

MENU ITEMS

The Menu Items for authoring 3D MicroWorld Exercises are:

File - This menu is used to close the viewer window.

Edit - This menu is used to define object Attributes, Enter scenes, and Cut/Copy/Paste/Delete objects.

Scene - The Scene menu is used to define New Scenes, Edit the background color of the current scene, and move to any defined scene.

Objects - This menu allows you to display the object Palette.

View - Switch between Examiner Viewer and Plane Viewer using this menu.

Options - The Texture Quality is set from this menu. Authors can also test the Scene from the student perspective.

Viewpoints - Authors create new viewpoints and move to stored points from this menu.

Help - Access the 3D Microworld Help file.

VIEWERS

There are two different viewers that can be used to develop microworld scenes. The default view is called the 'Examiner' view. The other available view is called a 'Plane Viewer'.

To switch between views, users select the 'Viewing' menu and choose the desired view.

Either viewer allows the user to interactively change the view of the scene through direct manipulation, or indirect slider and push button controls.

The Examiner viewer component (Figure 4) allows users to rotate the view around a point of interest using a virtual trackball. In addition to allowing camera rotation around the point of interest, this viewer also allows translating the camera in the viewer plane, as well as dolly (move forward and backward) to get closer to or further away from the point of interest. The viewer also supports seek to quickly move the camera to a desired object or point.

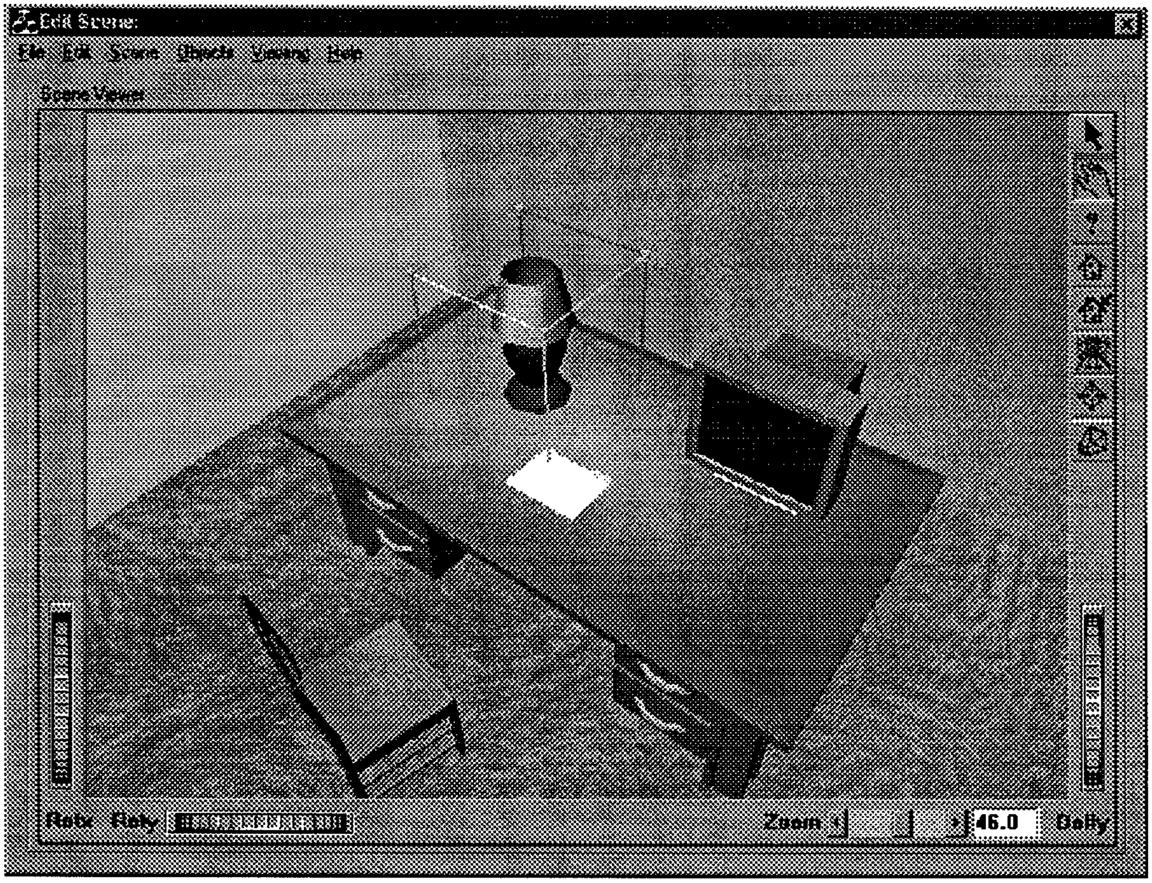


Figure 4 - Examiner Viewer

The Plane Viewer (Figure 5) constrains the camera to move only parallel to the view plane. Convenience buttons are provided (right side of viewer decorations) to set the camera in each of the major planes (X, Y, Z). The Plane viewer component allows the user to translate the camera in the viewing plane, as well as dolly (move forward/backward) and zoom in and out. The viewer also allows the user to roll the camera (rotate around the forward direction) and seek to objects which will specify a new viewing plane.

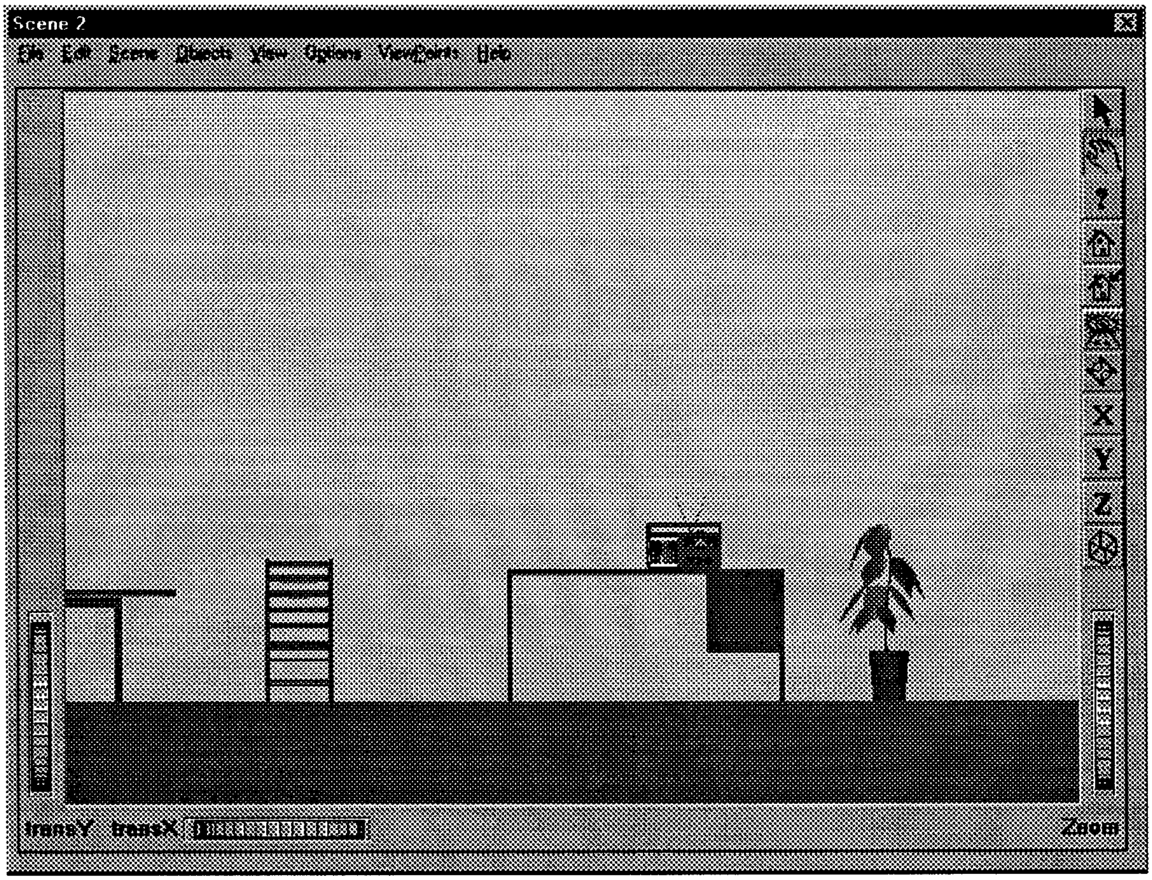


Figure 5 - Plane Viewer

EDIT SCENE TOOLS

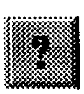
A toolbar is displayed to the right of the view. Each tool is described below. Note: Some of the tools are not available in both viewers.

 **Select/Pick Button**

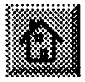
The author will use this tool to place objects in the scene, resize objects, rotate objects, and define object attributes. The cursor shape will change to an arrow.

 **View Button**

Selects viewer mode. The cursor shape will change to a hand icon. In this mode, the author can move the camera in 3D space.

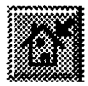
 **Help**

This menu provides help about the application.



Home Button

Returns the camera to its home position (initial position if not reset). Use this tool if you have "lost yourself" in the 3D environment.

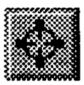


Set Home Button

Resets the home position to the current camera position.

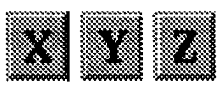
View All Button

Brings the entire scene into view.



Seek Button

Allows the user to select a new center of rotation for the camera. When clicked on (and in viewer mode) the cursor changes to a crosshair. The next left mouse button press causes whatever is underneath the cursor to be selected as the new center of rotation. Once the button is released, the camera either jumps or animates to its new position depending on the current setting of the seek time in the preferences dialog box.

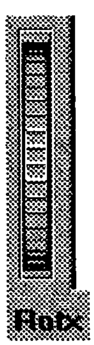


Camera Alignment Buttons

Select the axis of alignment (X, Y, or Z) of the camera. Use these buttons to change the currently displayed view. Only displayed for Plane Viewers.

Projection Button

It is used to select the type of camera used by the viewer. It toggles between the two available camera types --perspective and orthographic. Only displayed for Examiner and Plane viewers. The Dolly thumbwheel is only available to the perspective camera.

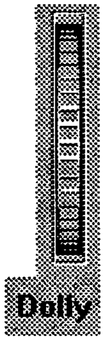


Rotate/Translate X Wheel

Rotates (Examiner View) or Translates (Plane View) scene about screen X axis.

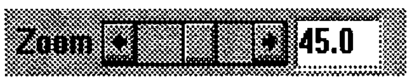
Rotate/Translate Y Wheel

Rotates (Examiner View) or Translates (Plane View) scene about screen Y axis.



Dolly Wheel

This wheel moves the camera in and out.



Zoom Slider

Adjusts the camera field of view. Field of view is specified in degrees.

MOUSE & COMMON CONTROLS

The 3D authoring environment uses a 'virtual trackball' to view the defined scene.

Control	Action
Left Mouse Button * View Mode Only (hand cursor)	Spin (rotate the virtual trackball)
Middle Mouse Button or <CTRL>+Left Mouse Button * View Mode Only (hand cursor)	Pan (translate up/down/left/right)
<CTRL> + Middle Mouse Button or Left Mouse Button + Middle Mouse Button * View Mode Only (hand cursor)	Dolly in and out
Right Mouse Button	Display the Viewer Popup Menu
<s> + Click * View Mode Only (hand cursor)	Seek Mode
<ALT>	Switch temporarily to Viewing Mode

Seek Mode - After pressing (and releasing) the <s> key, the cursor changes to a "target" shape. Click on the desired seek point with the left mouse button. After clicking on the seek point, the camera will be automatically moved to view the seek point.

Displaying the Viewer Popup Menu - The popup menu items allow you to change most of the viewer properties, such as drawing mode ("as is", "wireframe", etc.). There are also popup menu items corresponding to most of the viewer decoration buttons (Home, Set Home, View All, etc.).

Switching temporarily to Viewing Mode - When the viewer is in Selection mode, pressing and holding the ALT key temporarily switches the viewer to Viewing mode. When the ALT key is released, the viewer returns to Selection mode. Note: If any of the mouse buttons are currently depressed, the ALT key has no effect.

Adding New Objects

Objects are items that the author can place in a microworld scene and that students can manipulate during the exercise.

Authors can add new objects to the scene by:

1) Displaying the Object Palette (Figure 6) by selecting 'Palette' from the 'Object' menu.
2) Choosing an object to display from the list (Either double-click on the item or highlight it and click on the 'Apply' button)
3) Selecting the select tool (the arrow)
4) Clicking in the displayed view.

MILT will place the chosen object in the position where the mouse was clicked.

Collision detection was implemented so that authors can not place objects through existing objects. Authors can place objects on top of other objects, but cannot place them through any other object. New objects can be added to the outside "world" or to any scene.

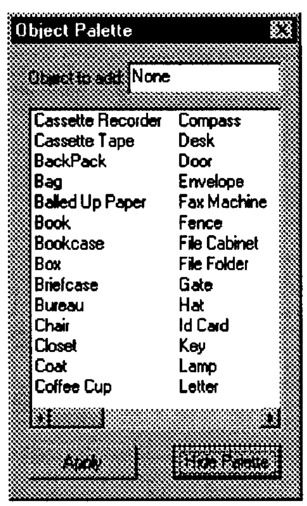


Figure 6 - Object Palette

The objects available are:

backpack	bag	balled up paper	book
bookcase	box	briefcase	bureau
cassette recorder	chair	closet	coat
coffee cup	compass	desk w/ drawers	door
envelope	fax machine	file cabinet	file folders
gate	hall	hanging picture	hat
id card	key	lamp	letter
map	newspaper	notebook	PA system
pants	paper	photo	plant
pop can	radio	secret compartment	secret door
shelf	shirt	shoes	sticky notes
table	tape cassette	television	tent
VCR	video tape	wall safe	wastebasket
white board	window		

Table 1 - Object List

<u>Changing the Object Size</u>

To change the size of an object:

1. click on the select tool (the arrow) for one of the displayed views
2. click on an object (the selected object will have a box around it displayed)
3. click on one of the square anchor points (the square will become highlighted)
4. click and hold the mouse while dragging the anchor until the desired size is obtained
5. release the mouse

The object size can only be made twice as big or 80% smaller than the default object size.

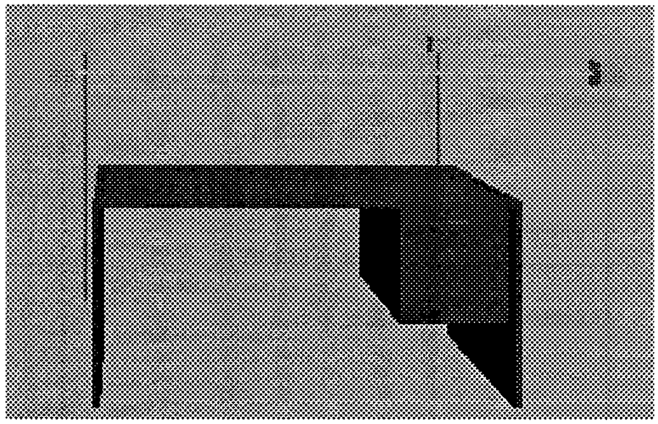


Figure 7 - Changing Object Size

Moving Objects within a Scene

To move a object within a scene:

1. click on the select tool (the arrow) for one of the displayed views
2. click on an object. A "selection" box will appear around it
3. click within the box. Arrows will appear indicating the plane of movement
4. click and hold the mouse on the object to move
5. drag the mouse to the newly desired position
6. release the mouse

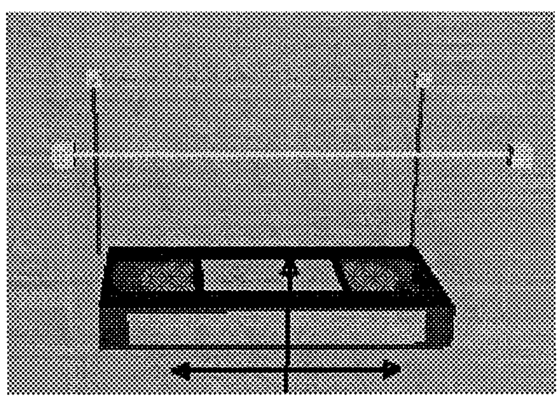


Figure 8 - Moving Objects

Rotating Objects

To rotate an object in the scene:

1. Click on the select tool (the arrow)

2. Click on the object to rotate (the selected object will have a box around it displayed)

3. Click on one of the lines surrounding the object (the line will become highlighted) Note: the line selected determines the rotation direction.

4. Click and hold the mouse while dragging the cursor until the desired rotation is obtained

5. Release the mouse

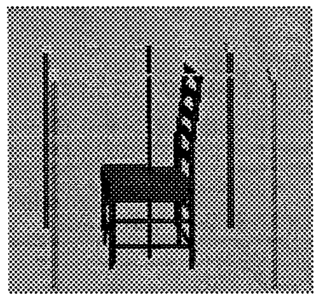


Figure 9 - Rotating Objects

Defining Object Attributes

Attributes are authorable components of objects. For example, text, color, sign, and open are attributes of a book object. For each book object placed in the microworld the author is able to change the text contained in the book, the size of the book, the color of the book, and the title of the book. The open attribute is also inherent in the book object (it can be opened or closed).

The authorable attributes available in Version 2.0 are:

<u>BMP</u> - the ability to display various sorts of bitmaps

<u>Color</u> - the ability to change object color

<u>Combination</u> - the ability to be opened with a 2 digit combination

<u>Connect</u> - the ability to link to another room when passed through

<u>Container</u> - the ability to put things into the object

<u>Hang</u> - can be placed on a wall

<u>Light</u> - the ability to change the visibility of a room when turned on.

<u>Lock</u> - the object has the ability to be locked.

<u>Open</u> - the ability to open as in a door, drawer, envelope, box, etc.

<u>Random</u> - it performs an action on its own at a random time after entry into room.

<u>Rotation</u> - the object can be rotated in 3 dimensional space

<u>Sign</u> - the ability to create text that appears on the object.

<u>Size</u> - the ability to be different sizes.

Size - the ability to be different sizes.

Text - the ability to author new text for that object.

Unlock - the object has the ability to unlock specifically assigned locked objects

Video - the ability to play defined digitized video file

WAV - the ability to play WAV files.

Authors can specify attributes for any object in the scene by double-clicking on the object. Each object has a unique set of authorable attributes.

Figure 10 shows what the object attributes window looks like. Only applicable attributes are shown for each object.

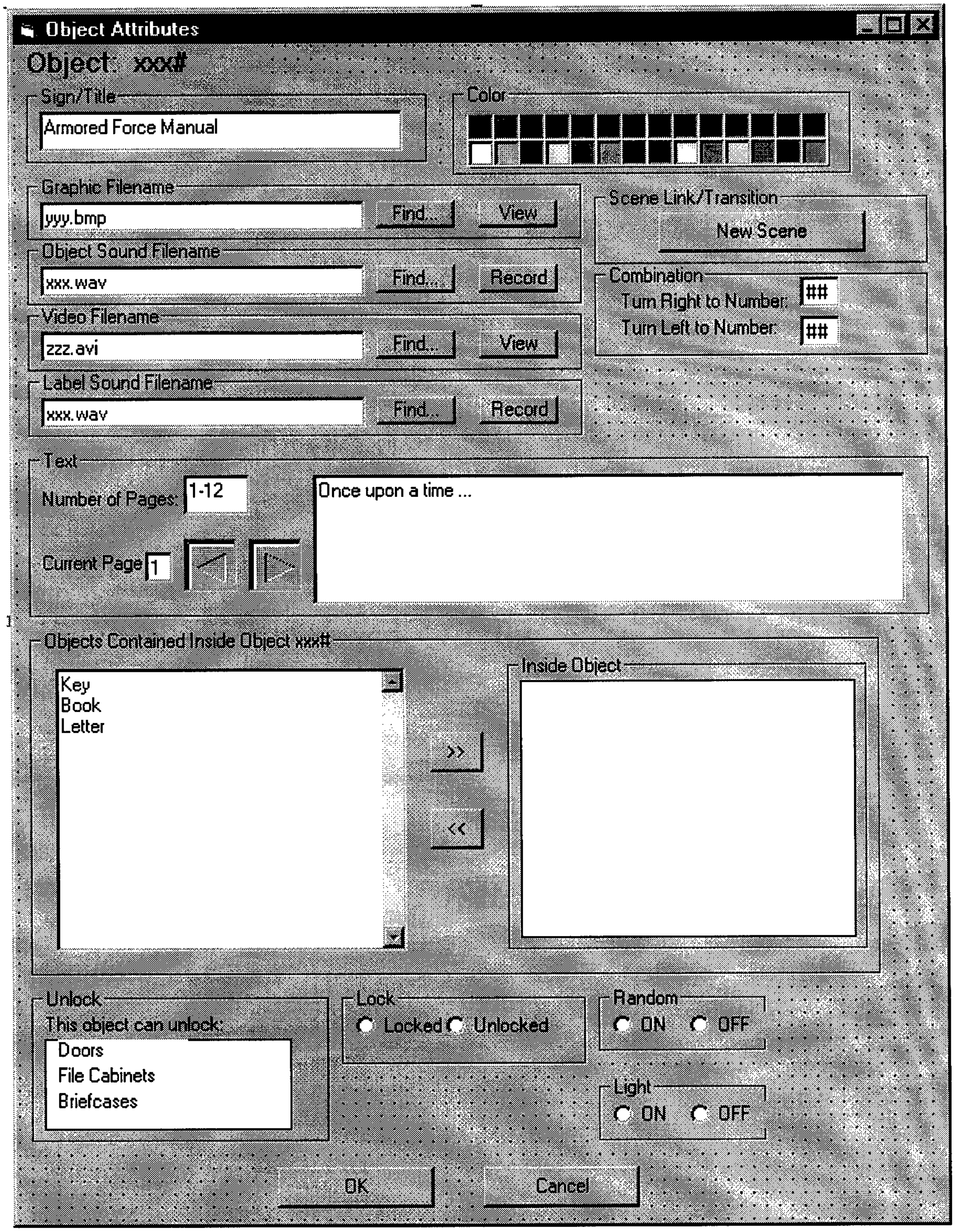


Figure 10 - Object Attribute Window

The following table lists the attribute, how the author can change it, and any constraints on defining the attribute.

Attribute	Objects w/ Attribute	Authoring	Constraints
BMP	fax machine, id card, map, photo, picture	Enter name of bitmap file into the graphic filename field	graphic must be bitmap (.bmp)
Color	All	Select color from palette	10 colors
Combination	wall safe	Enter a 2 digit number in the 'Turn Right to Number' field and a 2 digit number in the 'Turn Left to Number:' field.	Numbers must be between 00 and 10
Connect	door, window, secret panel	Click on "New Scene" button; pick scene type (room, outside, etc.)	Can only be placed from within scene
Container	backpack, bag, bookcase, box, briefcase, cassette recorder, closet, coat, coffee cup, desk, envelope, file cabinet, file folder, pants, pop can, secret panel 1, shirt, shoes, VCR, wall safe, wastebasket	Click on object representation of object to be placed inside, click in the 'inside object' view. MILT will place object where clicked. Double click on object displayed in 'inside object' view to edit.	Only objects that are smaller than container object can be placed inside it. Cassette recorder and VCR can only have appropriate tapes placed inside.
Hang	picture, PA system, shelf, white board, wall safe	Place any of these objects on a wall and they will stay	Any object besides those listed here will fall to floor when placed on a wall
Label	All	enter synonyms	none

Attribute	Objects w/ Attribute	Authoring	Constraints
Label Sound File	All	Enter the name of desired file into the Sound Filename field or record sound file	Must be WAV format. User computer must have sound card.
Light	lamp	Select either the On or OFF radio button to determine initial state	none
Lock	briefcase, door, file cabinet, wall safe	Select either Locked or Unlocked to set initial state	The wall safe is opened by a two number combination; all other objects are opened with a key.
Open	backpack, bag, balled up paper, book, box, briefcase, cassette recorder, closet, desk, door, envelope, file cabinet, file folder, gate, map, newspaper, notebook, secret panels, VCR, wall safe, wastebasket, window	None	Select either Open or Closed
Random	fax machine, PA system	Select ON to activate or OFF to deactivate	Due to random nature may or may not be triggered while student is in the room

Attribute	Objects w/ Attribute	Authoring	Constraints
Rotation	All	Select desired angle and dimension from Object Rotation field	Supported dimensions are horizontal and vertical. Supported rotation angles are 90°, 180°, and 270°. No intermediate rotation is allowed.
Sign	bag, book, bookcase, cassette tape, coffee cup, door, file cabinet, file folder, key, notebook, video tape	Enter text in Sign/Title text field	maximum of 64 characters
Size	All	Enter scale factor in percentage	All objects are scaled to be of the correct relative size. Smallest size 50%, largest 200%.
Text	balled up paper, book, envelope, fax machine, id card, letter, newspaper, notebook, paper, sticky note, white board	Click in the Text field and enter text using the keyboard	All objects except the book have 1 text field. The book has two text fields.
Unlock	key	Select 1 or more classes of objects that the key can unlock from the Unlock field	Keys will not be specific to any individual objects, but will be ties to classes of objects. For example, the blue key might unlock all doors, the red key all briefcases, and the gold key all locked objects.

Attribute	Objects w/ Attribute	Authoring	Constraints
Video	television, video tape	Enter the name of file into the Video Filename field	Must be avi format. User computer must support video formats.
Sound	cassette tape, PA system, radio	Enter the name of desired file into the Sound Filename field or record sound file	Must be WAV format. User computer must have sound card.

Table 2 - Objects and Attributes

Defining objects that are inside another object

Objects that can have other objects placed inside them are called Container Objects. To place objects inside another object:

1. Double-click on an object
2. MILT will display the object attribute window
3. From the list of items that can fit in the container, select item(s) to place inside and click arrow to move object(s) inside container item
4. Click the OK button

To edit the attributes of any object that is within the container double click on the object name from the 'Objects Inside Container' list.

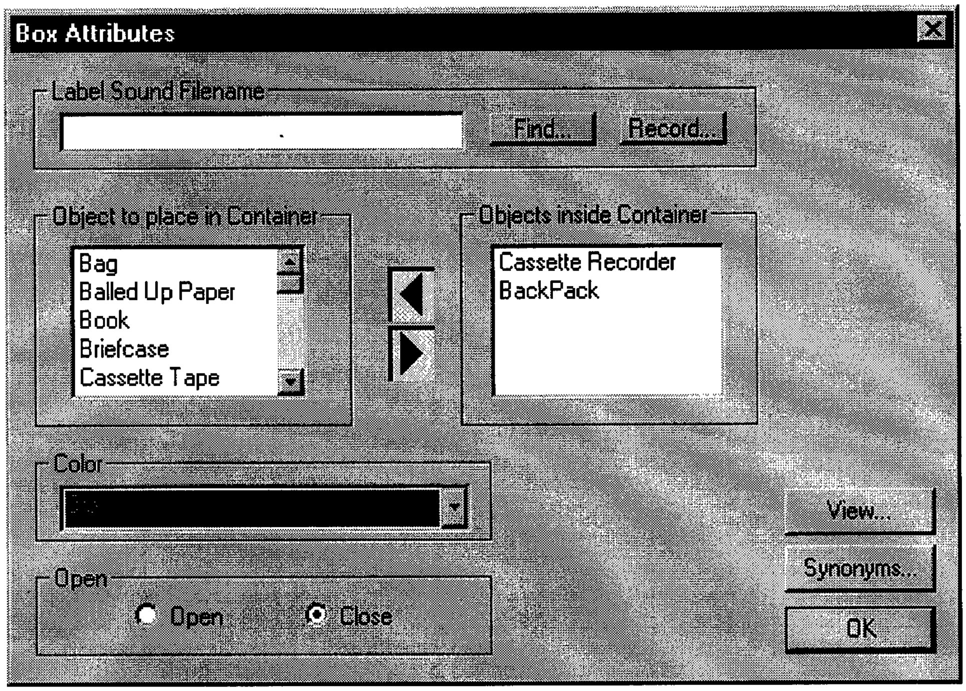


Figure 11 - Example Container Attribute Window

Defining a New Scene

When the author double clicks on a scene representation for a scene that does not yet have any characteristics or objects defined, MILT will prompt the user to specify the type of scene to be added. The screen looks similar to **Figure 12**.

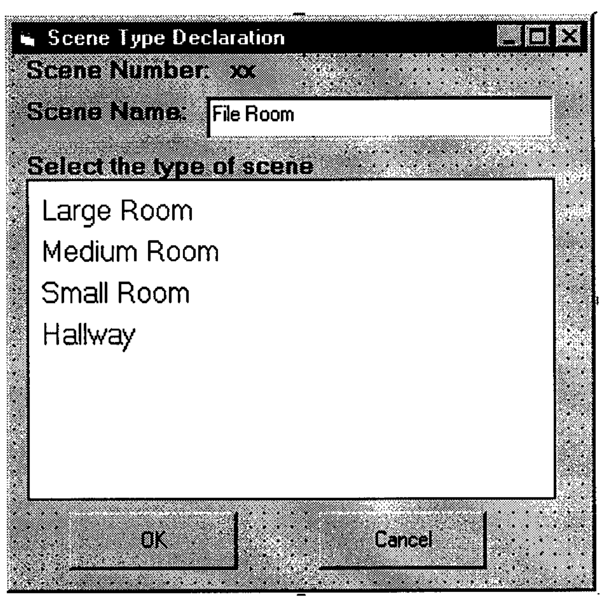


Figure 12 - Scene Type Declaration

MILT automatically assigns a scene number to the new scene. The scene type defines the dimensions inside the new scene. MILT 2.0 provides different scene types of predefined size for

the author to chose from. The different scenes range in size and shape. The author will place the selected building type on the outside scene.

Editing/Defining Scene Characteristics

Scene attributes are name and color.

To edit scene attributes:

1) Click on the Scene with the Select Tool (the arrow)

2) Select 'Attributes' from the 'Edit' menu

3) Click in the 'Name' field and enter the scene name and/or

4) Click on the color list and select one of the displayed options

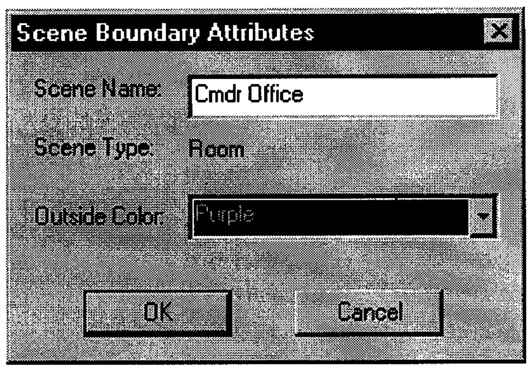


Figure 13 - Scene Characteristics

Editing Scene Background Color

The background color for any scene can be specified by:

1) Selecting 'Edit Background Color' from the 'Scene' menu

2) Selecting a color from the color wheel or slider bar(s)

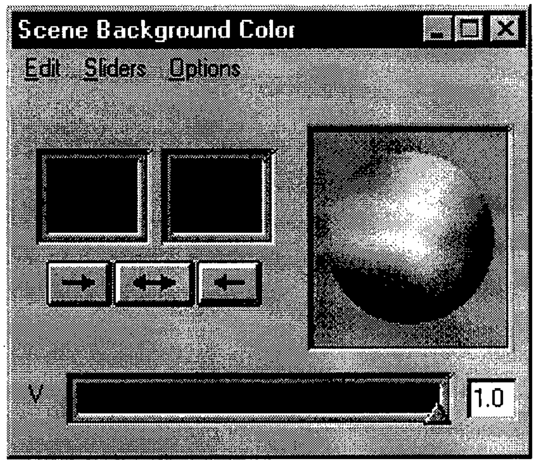


Figure 14 - Scene Background Color

The background color chosen for the Outside "world" is displayed as the sky and space around the view. Background colors for internal scenes are only seen as the space outside the scene boundaries.

Entering a Scene

Once a scene has been defined in the Outside "world" authors can enter the scene. Within any scene the author can place new objects and define object attributes.

There are two methods that can be used to enter a scene.

To use the first method:

1) Click on the scene representation with the Select tool (the arrow)

2) A square selection box will appear around the scene

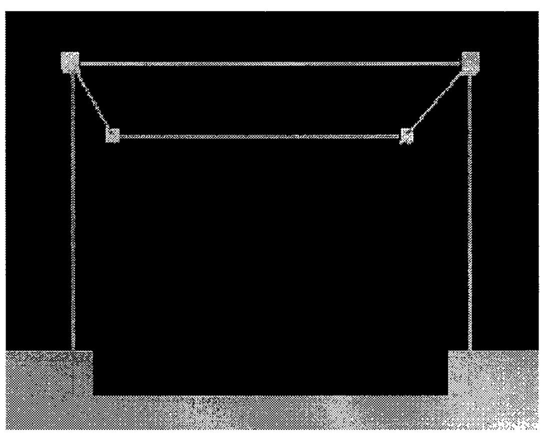


Figure 15 - Select Scene

3) Select 'Enter' from the 'Edit' menu

The second method involves selecting the scene name to enter from the 'Scene' menu

Editing Scene Boundary Attributes

Boundaries are the "edges' of each scene. The dimensions of the boundaries are determined by the scene type (room, outside, corridor, etc.) and cannot be changed by the author.

Authors can change the "look" of the boundary by applying a background texture from the MILT library or applying a custom texture to the wall.

To define the boundary attribute:

1) From within any scene, double click on a boundary from the Edit Scene Window.

2) MILT will display the boundary attribute window. (Figure 16)

First, choose which boundary to edit by selecting the name from the 'Boundary' list. Then, the texture, color, and/or texture scale of that boundary can be changed.

Change the texture by clicking on the 'texture' button and selecting from the list (brick, wood, light carpet, cobblestone, sky, grass, tile, stucco).

Change the color of the selected boundary by moving the Red, Green and/or Blue slider bars.

Change the texture scale by clicking on the appropriate horizontal and/or vertical selection buttons. The texture scale refers to the number of horizontal and vertical tiles of the texture are applied to the boundary. MILT comes with reasonable default settings for each texture. Note: if a texture is not selected this option will not work.

Light intensity slider bars allow the author to set the room brightness.

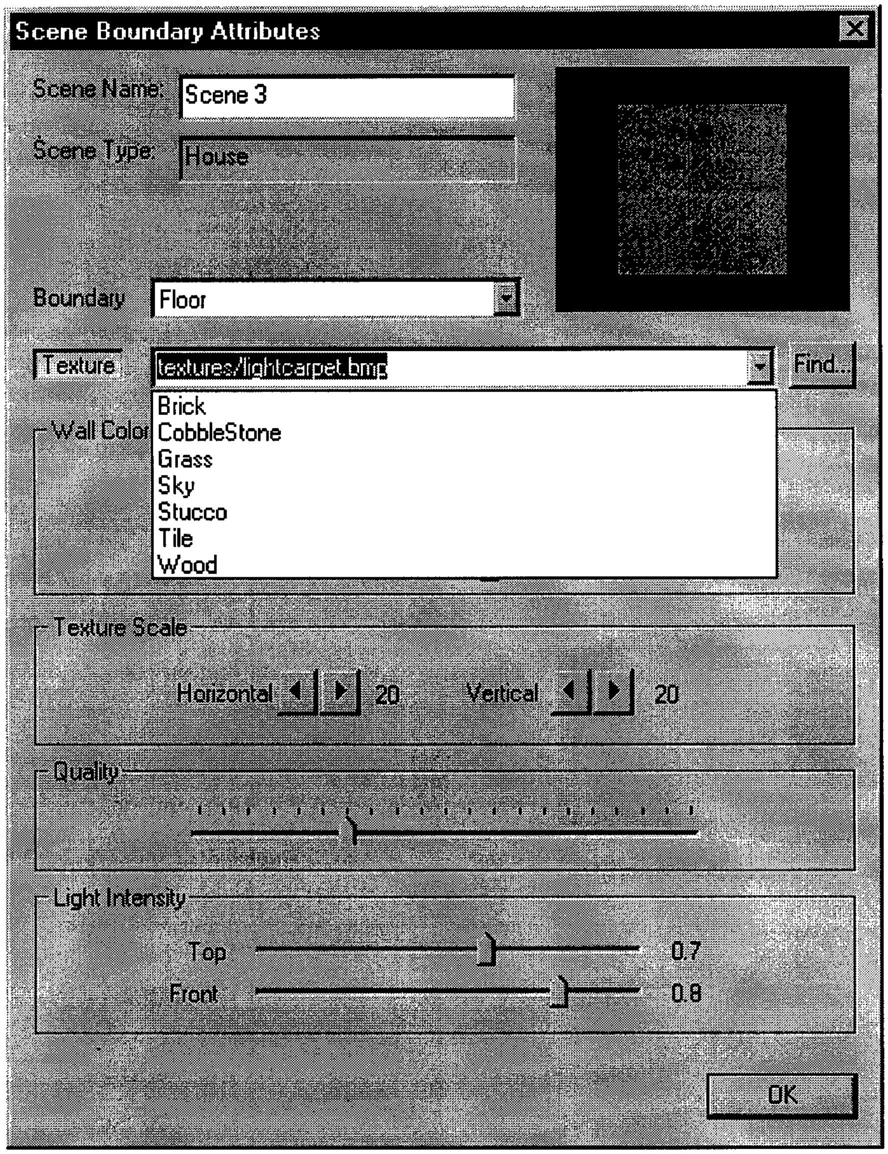


Figure 16 - Scene Boundary Attributes

Connect/Transition Attribute

Certain objects (doors, windows, and secret panels) allow the student to transition from one scene to another by going through the object.

To link scenes together, the author must

1) Define the scenes in the Outside "world" (See Defining A New Scene)
2) Enter one of the scenes (See Entering a Scene)
3) Add a transition object to the scene (See Adding New Objects)
4) Click on the object with the Select Tool (the arrow)

5) A selection box will be displayed around the object

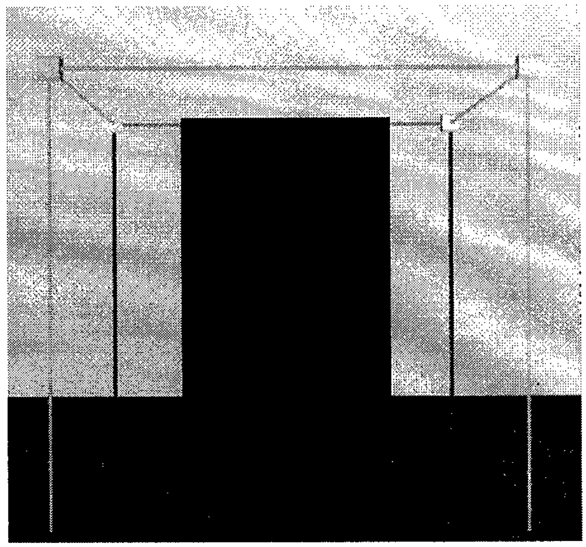


Figure 17 - Selection Box

6) Move the object onto one of the boundary walls
7) Select 'Attributes' from the 'Edit' menu
8) Specify the scene the object connects to from the 'Connected to Scene' list box
9) Click on 'OK'

Setting Texture Quality

The texture quality affects the speed that the microworld exercise executes. The higher the texture quality the slower the speed.

To set the texture quality:

1) Select 'Texture Quality" from the 'Options' menu
2) Enter a value between 0.0 and 1.0 (recommended value is 0.5)
3) Click on the 'OK' button

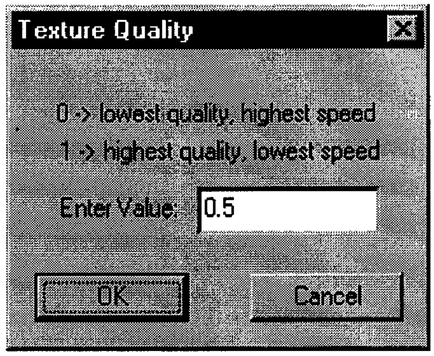


Figure 18 - Texture Quality

Using Viewpoints

Viewpoints are essentially anchor points that authors may want to return to when defining a scene. By setting viewpoints the current camera position is stored.

To set a viewpoint:

1) Move around the scene using the viewer controls until the camera is positioned where you want it
2) Select 'Create Viewpoint' from the 'Viewpoint' menu
3) Enter a name for the viewpoint that describes the current position
4) Click the 'OK' button

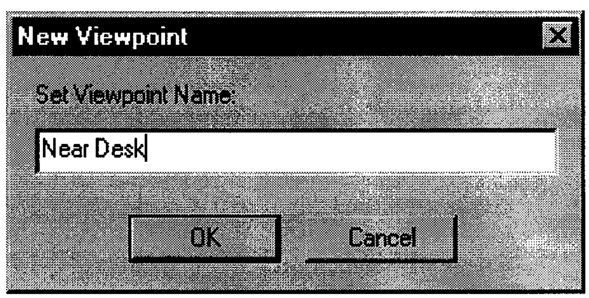


Figure 19 - New Viewpoint

Moving to a Viewpoint:

1) Click and hold the mouse on the 'Viewpoint' menu

2) Move the mouse down until the desired Viewpoint name is highlighted

3) Move the mouse to the left until 'Seek' is highlighted

4) Release the mouse

The view will change to the stored viewpoint.

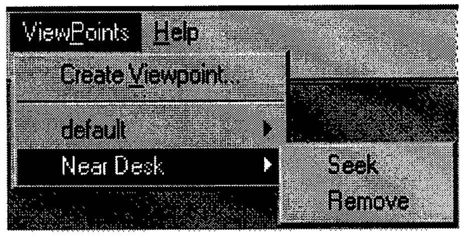


Figure 20 - Viewpoint Menu

Removing to a Viewpoint:

1) Click and hold the mouse on the 'Viewpoint' menu

2) Move the mouse down until the desired Viewpoint name is highlighted

3) Move the mouse to the left until 'Remove' is highlighted

5) Release the mouse

Test Scene/View Exercise

Once a microworld scene has been defined, the author can test it from author mode. This will allow authors to view the newly created scene from the perspective of the student.

In the student perspective, users issue commands to move around the microworld and manipulate objects. The goal is to find the answer to a question. Students will see a window with a 3D microworld display area and command entry area. To test the exercise, issue commands to search the scene or scenes for clues to answer the question.

There are two possible modes of input for the microworld. In one mode, you can issue commands by typing into the text field. In the second mode, you can issue commands by speaking into a microphone. Switch between modes by clicking on the Talk checkbox.

There are only certain actions that are recognized in the microworld. If the command issued to the microworld is recognized you will see an action occur in the microworld display.

The Microworld Exercise from the Student Perspective

Overview

In the microworld exercise, the student sees a window with a 3D microworld display area and command entry area similar to Figure 21. In this exercise, the student issues commands to search the scene or scenes for clues to the answer to the question. There are two modes of the microworld. In one mode, the student issues commands by typing into a text field. In the second mode, the student issues commands by speaking into a microphone. When the student feels that he has the correct answer to the question, he will select 'Check Answer' from the Exercise menu and type the suspected answer into the space provided. To review the question, the student selects 'View Question' from the Exercise menu.

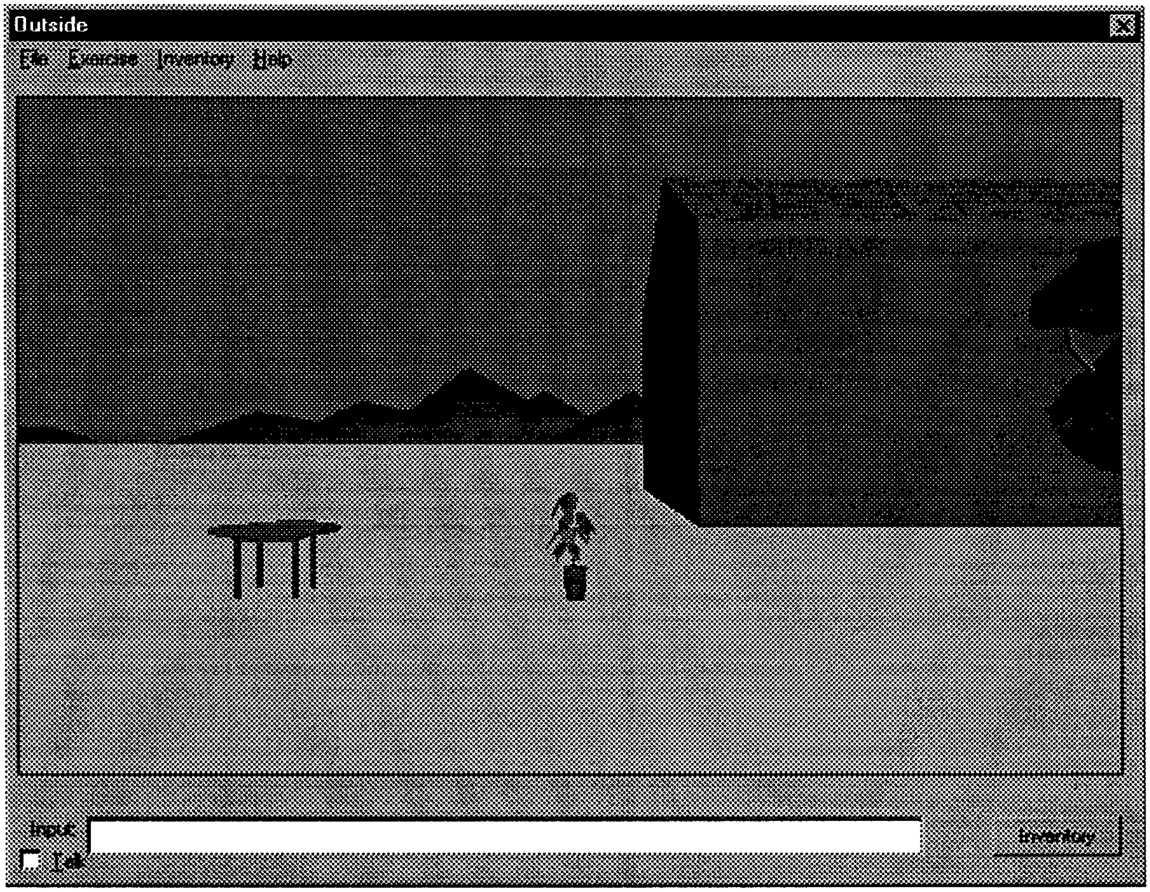


Figure 21 - Microworld Exercise

<u>Actions</u>

The microworld exercise will only allow a predefined and limited set of actions to occur. The actions that are available in Version 2.0 are:

carry the _____

carry the ____ to the ____

climb the ____

climb on the ____

climb off the ____

climb over the ____

climb under the ____

climb through the ____

close the _____

cover the ____ with the _____

crawl through the ______

crawl under the ______

drop the ______

eat the ______

enter the _____

get down from the ____

get off the ______

get the ____

get the ___ from the _____

get the ___ from inventory

get up

give the ___ to the ____

go back

go in the ____

go left

go East

go North

go South

go West

go right

go through the ____

go to the ____

hang the picture on the wall

hear the _____

hit the ____

hold the ____

insert the ___ into the ____

insert the ____

jump over the _____

jump down

jump off the _____

jump on the _____

jump to the _____

kick the ____

knock on the ____

leap over the ___

leap onto the _____

leave the ____ on the ____

leave the ____

left to (followed 2 digits)

lift the ___

lift up the ____

listen to the ____

load the audio cassette

load the video cassette

lock the ____

look at the ____

look behind the _____

look down

look in inventory

look up

look to the left

look to the right

look in the ____

look in your_____

look under the ____

look North

look South

look West

look East

move all but the ____

move forward

move left

move right

move the____

move the ____ to the _____

move the ____ under the ____

move the ___ from the ___ to the ____

move the ___ behind the_____

open the _____

pause the _____

pay the man

pick up the ______

place the ____ on the ___

place the _____ under the ____

place the____ behind the _____

play the _____

put down the _____

put the ____back

put the ___ in inventory

put the cassette in the cassette player

put the cassette in the VCR

put the ______ down

put the ____ on the _____

put the ___ under the ______

put the ___ in the ____

read the _____

remove the ______ from inventory

retrieve the _____

right to (followed by 2 digits)

run to the ____

search for ____

search the ____

see the____

set the ____ on the ____

set the ____ under the _____

show the ____

sit

sit on the ___

sit under the ____

stand

stand on the ____

start the _____

steal the ____

stop the _____

take the ____

take the ____ out of ________

take the ____ from inventory

take the picture down

taste the ____

the ___ goes into the _____

throw the_____

touch the____

turn around

turn left

turn right

turn off the_____

turn on the _____

turn right

turn the key

unfold the _____

unroll the ___

unlock the ____

walk forward

walk left

walk right

walk North

walk South

walk East

walk West

walk straight ahead

walk to the ______

walk to the end of the hall

walk through the_____

If the command issued to the microworld is recognized the student will see an action occur in the microworld display.

"Look and Feel"

THE FIRST PERSON VIEW

The first person view and 3D nature of the microworld enables the view to change as the student issues commands. The students view is essentially like a camera moving around 3D space. The camera can zoom in, zoom out, rotate horizontally, and rotate vertically. For example, if a student issues the command "walk to the table" the student sees the table become closer and closer in the view until the table is reached.

REACHING

If the student reaches for an object that will go into inventory the object will get closer and closer in the view then disappear.

If the student picks up an object and carries it to another location, the object will get closer and closer in the view, the camera will travel to the "go to location", and the object will move father from the view.

Example: if a book is on the chair and the student issues the command "carry the book to the table." The camera will move to the chair, the book will move to the camera (get bigger), then the book will disappear from view, the camera will move to the table, the book will appear close, then the book will move onto the table (getting smaller).

Body parts (arms, hands, feet, etc.) of the 1st person entity (the student) will not appear.

Moving Between Scenes

When a student goes into a new scene and turns around, he will see a boundary scene with the correct number and type of transition objects (windows, doors, etc) connecting to the new scene displayed.

Labels

Students will have the ability to gain information about objects by looking at that objects "label". The label will display the name of the object and give the student information about what actions can be performed on it. Labels will be displayed when the user selects an object with the mouse. The label will look similar to Figure 22. The text on the 1st screen will appear in English. If the exercise was constructed in a foreign language, the student will have the option of also viewing the label information displayed in that language by clicking on the "Foreign Language" button. The student will also be able to listen to a WAV recording that was made by the instructor. The Foreign Language display will be created from the synonym lists generated by the author.

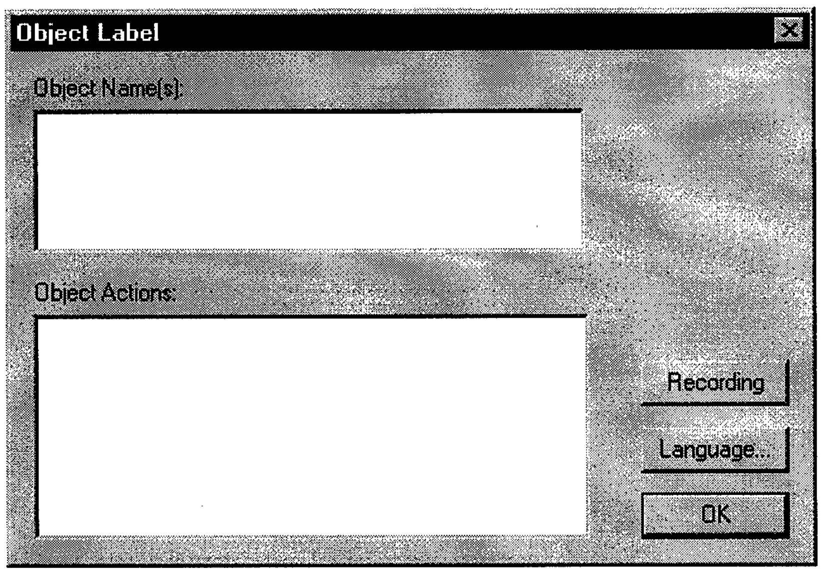


Figure 22 - Label Window

Object Inventory

The ability to inventory objects is required in order to allow the student to carry multiple objects from place to place and for the student to pick up multiple items. The inventory functionality was implemented using a combination of automatic text recognition and the "inventory" button on the screen.

PLACING OBJECTS INTO INVENTORY

MILT automatically places objects in inventory whenever the student issues one of the following commands:

carry the _____

hold the ___

get the ____

pick up the ____

lift the ___

lift up the ___

retrieve the ___

steal the ___

take the ___

get the ___ from the ___

get the ___ out of the ___

pick up the __from the ___

pick up the __out of the ___

lift the ___ from the ___

lift the ___ out of the ___

lift up the ___ from the ___

lift up the ___ out of the ___

retrieve the ___ from the ___

retrieve the __ out of the ___

steal the ___ from the ___

steal the ___ out of the ___

take the ___ from the ___

take the ___ out of the ___

put the ___ in/into inventory

place the __ in/into inventory

move the __to/into inventory

VIEWING OBJECTS IN INVENTORY

The collection of items in inventory can be viewed by the student either through a command or a button push. Once either method is invoked, MILT will display a window that shows a list of all objects currently in inventory.

The command required to view inventory items are:

look in inventory

check the inventory

view inventory

The student can also view inventory contents by clicking on the 'Inventory' button.

REMOVING OBJECTS FROM INVENTORY

Items can be removed from inventory in two ways. The first method that can be used to remove an item from inventory is through the use of commands. The following commands will remove items from inventory:

Get the ___ from inventory

Remove the _ from inventory

Take the ___ from inventory

Retrieve the _from inventory

Move the ___ from inventory

Put down the ____

Put the ___ down

The second method that can be used to remove items from inventory is by clicking on the inventory button and selecting the item from the inventory list.

Once the student has selected an item to remove, MILT places that item on the ground in the current scene.

The student must remove an item from inventory before he can use it. For example, if a map is in the inventory and the student wants to look at it, he will need to remove it from inventory.

Attribute Display

Various objects in the microworld have different attributes as shown in Table 2. The student triggers various attributes to be changed or displayed by issuing appropriate commands.

TEXT DISPLAY

Objects that have the authorable text attribute are: balled up paper, book, envelope, fax machine, id card, letter, newspaper, notebook, paper, sticky note, and white board. The student can access the text by issuing one of the following commands:

read the <<object>>

look at the <<object>>

Once the command has been issued MILT brings up a text display window. The background for the authorable text will be dependent on the object and will be a realistic representation of the object.

The book and notebook can have a maximum of 12 pages displayed. All other objects have only one text field. The student turns the pages by clicking on the corner with the upturned page showing.

CONTAINED ITEM DISPLAY

There are several objects that can contain other objects. These "container" items are: backpack, bag, bookcase, box, briefcase, cassette recorder, closet, coat, coffee cup, desk, envelope, file cabinet, file folder, pants, pop can, secret panel 1, shirt, shoes, VCR, wall safe, and wastebasket.

The student is able to see the contents of an object using the following commands:

open the _____

look in the ______

MILT 2.0 displays the contents in the same manner as MILT 1.x – once the student enters one of the above commands, MILT will "pop out" all of the objects that are contained inside.

VIDEO DISPLAY

The television and VCR can play video files. The playing of the video are triggered by the following commands:

turn on the (VCR, television)

play the VCR

The video window does not appear in the object. It appears in a separate window than the microworld scene.

Handling Multiple Objects of the Same Type

Version 2.0 allows multiple objects of the same type to be used in a single scene and allows the student to carry objects into another scene that may already have the same type of object in it. This capability adds complexity to the microworld.

If more than one object of a given type exists in the scene, MILT uses the following rules to determine which object is being acted upon (in order of priority).

1. Determine object using differentiating adjectives (red, blue, big, ...)
2. Use in object displayed in current view
3. Use last direct object named
4. Use object closest in proximity

Continuous Speech Recognition

In the speech recognition mode the student utters commands into the microphone instead of typing them in using the keyboard. As the student speaks, MILT displays in text what the recognizer thinks the student is saying.

Task 7: Develop Arabic CSR models

The speech recognition components were developed using the HTK application developed by Entropic Cambridge Research Laboratory. How HTK achieves recognition is best understood in terms of the structure of a speech recognizer illustrated in Figure 23.

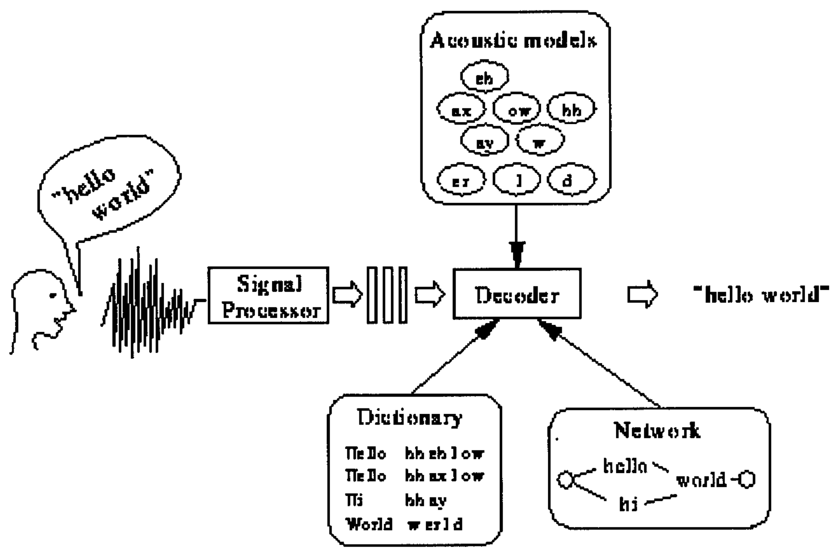


Figure 23 - Speech Recognition Components (Ollason, 1997)

The central element of a speech recognizer is the decoder. This transcribes continuous speech input into a textual symbol sequence that an application can directly process. It requires a syntax network to define the allowed word sequences appropriate for the recognition task, a

pronunciation dictionary specifying an acoustic model sequence for each word in the task vocabulary and a set of acoustic models which model the individual speech sounds.

The HTK system developed by Entropic includes a dictionary, decoder and a set of pre-trained acoustic models for English and Spanish. However, Arabic modules did not exist.

John Morgan and Steve LaRocca were the principle developers at the U.S. Military Academy (USMA) who created the phone-level speech acoustic hidden Markov models (HMMs) for Arabic. This section describes the steps taken by personnel at the USMA to develop the components for the Arabic speech recognizer. USMA followed the example given in Entropic's Documentation (Odell, 1997) to build the acoustic models for Arabic and a language model for MILT's Microworld commands. The steps that were followed are as follows:

Step 1. Data Preparation

- 1.1 Creating the Prompt Scripts
- 1.2 Creating the Dictionary
- 1.3 Recording The Data
- 1.4 Creating the Transcription Files
- 1.5 Coding The Data

Step 2. Creating Monophone HMMs

- 2.1 Creating Flat Start Monophones
- 2.2 Fixing the Silence Models

Step 3. Creating Tied-State Triphones

Step 4. Creating The Language Model

Step 5. Recognizer Evaluation

Each of the steps is described in the remainder of this section.

Step 1: Data Preparation

1.1 Creating the Prompt Scripts

One of the U.S. Military Academy Arabic instructors, Mr. Raja Chouairi, wrote two prompt scripts. We used the first script to collect data from 58 native Arabic speakers. The script contains 155 prompts. The lines in the script roughly correspond to lines spoken by actors in a play. Two characters meet in a cafe, where they discuss topics such as family, food, religion and relationships. The script has a total of 1152 words 724 of which are distinct. We define a ``word\'\' as the set of characters delimited by white space. So we consider ``wa-man\'\' (``and who\'\' in English) to be a single word. We used the second script to collect data from 25 nonnative Arabic speakers. This script only contains 40 short sentences and covers topics similar to the first script. The nonnative script has a total of 150 words 124 of which are distinct. Mr. Chouairy wrote both scripts in Modern Standard Arabic (msa). He originally wrote the

script in a 7-bit ascii dialect of LaTeX called arabtex. This facilitated processing data with HTK. Later he re-wrote the scripts as a Wincalis[1] script in Unicode. The WinCalis script contains both a textual prompt in the Arabic calligraphy and an auditory prompt.

The scripts are shown below (in ArabTeX format, See Appendix A for ArabTeX conventions).

The first script is as follows:

1. mar.habaN mA \'ismuka
2. \'ahlaN \'ismy rajA wa-man \'anti
3. \'anA rImA \'anA mi.sriyyaT min \'ayna \'anta
4. \'anA min lubnAn \'anA lubnAny
5. mA_dA ta`mal yA rajA
6. \'anA .tAlib wa-\'anti mA_dA ta`malIn
7. \'anA mudarrisaT \'ayna taskun yA rajA
8. \'askun fy ha_dihi al-min.taqaT bi-al-qurbi mina al-^gAmi`aT
9. baytuka qaryb mina al-^gAmi`aT walakin manzily ba`yd `anhA
10. fy al-.haqyqaT lA \'a`y^s fy bayt wa lakin fy ^siqaT
11. mA_dA tadrus yA rajA
12. \'anA \'adrus al-ttAry_h wa-al-^gu.grAfiA
13. ha_dA ^gayyid ^gidaN hal tadrus ka_tyraN
14. na`am \'anA dA\'imaN ma^s.gUl mina al-.s.sabA.h \'il_A al-masA\' wa-\'anti
15. \'anA \'udarris al-.t.tib fy nafsi al-^gAmi`aT wa-\'anA ma^s.gUlaT \'ay.daN
16. \'anti .tabybaT \'aby huwa .tabyb wa-\'ummy hiya `AlimaT \'i`lAm
17. \'ummy hiya rabbaT bayt ta`mal kul al-waqt
18. hiya ma^s.gUlaT ka_tyraN fy al-bayt .hay_tu ta`mal
19. kam lu.g.gaT tatakallam bi-\'isti_tnA\'i al-lu.g.gaTi al-`arabiyyaT
20. fy al-.haqyqaT _talA_taT wa\'anti kam lu.g.gaTaN tatakallamyn
21. \'a`rif `A\'ilataka min al-.s.sUraT al-^gamylaT
22. hal ta`rifyn `A\'ilaty mina al-rrisAlaTi al-.t.tawylaT \'am al-qa.syraT
23. lA walakin \'a`rif zaw^ga _hAlatika ^gAlis \'il_A al-yamyn
24. wa-ha_dihi _hAlaT wAlidy ^gAlisaT \'il_A al-yasAr

[1] WinCalis is a Windows Based Computer-Assisted Language Instruction System developed by Humanities Computing Facility, Duke University.

25. wa-ha_dA `am wAlidatika wAqif warA\'a zaw^gaT \'a_hyka al-wAqifaT \'ay.daN

26. na`am huwa .dAbi.t fy al-^gay^s mi_tla \'a_hyh

27. mi_tl \'ibn `ammy wa-bint _hAly wa-\'a_hyhA

28. kullu \'aqAriby hum fy al-^gay^s walakin laysa ma`ahum fulUs

29. fy \'ay ^gay^s ya_hdumUn

30. fy al-^gay^s al-lubnAny li-\'anna lubnAn huwa wa.tanuhum

31. wa-al-wa.tana al-`araby huwa wa.tanunA

32. tab`aN hal `indaki waqt al-\'An

33. lA laysa `indy waqt al-\'An limA_dA tas\'al

34. \'as\'al li-\'annany \'uryd \'an \'ad`Uki \'il_A fin^gAn qahwaT

35. sa\'aqbal al-dda`waT walakin lA \'a^srab al-qahwaT

36. rubbamA ta\'_hu_dyna ka\'asa ^sAy wa-kubbAyaTa mA\'iN

37. sa\'aqbal al-da`waT walakin \'intabih \'anA mutazawwi^gaT wa-ly \'a.hsan ra^gul fy al-`Alam

38. wa-\'anA \'a`zab wa-laysa ly \'imra\'aT \'anA \'a_dk_A ^sa_h.s fy al-`Alam

39. \'a^s`ur \'annaka ^sa_h.suN wa-.hyduN wa-mina al-llAzim \'an ta^gida linafsika rafyqaTaN li-.hayAtik

40. rubbamA yA .gArsUn qahwaT wa-^sAy min fa.dlika

41. liman al-^s^sAy wa-liman al-qahwaT yA \'ustA_d

42. al-qahwaT ly wa-al-^s^sAy lahA wa-ha_dA al-ba_h^sy^s laka

43. lA ta^srab al-^s^sAy al-ssA_hin yA ra^gA

44. na`am walakin `inda al-fu.tUr faqa.t bayna al-.gadA\' wa-al-`a^sA\' \'ufa.d.dil al-qahwaT

45. hal tu.hib al-^s^sAy al-\'amryky al-mu_talla^g

46. tab`aN \'u.hibbuhu ka_tyraN walakin fy fa.sli al-.s.sayf li-\'anna al-.t.taqs .hAr

47. wa-fy fa.sl al-^s^sitA\' `indamA yakUnu al-^gawwu bAridaN tufa.d.dil al-ma^srUb al-ssA_hin

48. bi-al-.t.tab` wa-fy al-rraby` wa-al-_haryf \'a^srab al-ma^srUbAt al-ku.hUliyyaT _hA.saTaN ma`a al-\'akl

49. \'anA lA \'a^srabahA li-\'annany muslimaT mA huwa naw`u \'aklika al-mufa.d.dal

50. ta`Amy al-mufa.d.dal huwa mina al-ma.tba_hi al-llubnAny wa-\'u.hibbu al-mulU_hiyyaT ka_tyraN

51. wa-lakin yA ra^gA al-mulU_hiyyaT huwa .tabaquN min \'a.sliN mi.sry

52. `a^gyb ^gaddaty lam taqul ly ha_dA mun_du .tufUlaty wa-hiya tu.ha.d.diruhA ly

53. mA qAlat laka ha_dA rubbamA li-\'annahA lubnAniyyaT hal ta`y^s ^gaddatuka fy bayt \'ahlik
54. \'awalaN ^gaddaty laysat lubnAniyyaT _tAnyaN \'intaqalat \'il_A baytinA
55. ba`da wafAT ^gaddy al-ssanaT al-mA.diyaT
56. al-ll_ah yar.hamuhu yabdU \'anna hiwAyataka al-mufa.d.dalaT hiya al-\'akl
57. wa-li`bu al-waraq wa-tad_hynu al-\'ar.gylaT wa-saharu al-llayl ma` al-\'a.sdiqA\' sykAraT
58. ^sukraN lA \'uda_h_hin \'ammA hiwAyAty fa-hiya al-mu.tAla`aT wa-qirA\'T al-^garydaT
59. wa-mu^sAhadaT al-ttalfiziyUn
60. \'anA lA \'u.hibbuhu walakin \'ufa.d.dil mu^sAhadaT al-nnAs wa-al-nna.zar \'il_A al-.t.taby`aT
61. mA_dA tar_A fy al-.t.taby`aT wa-al-nnAs .gayr al-.huzn wa-al-ma^sAkil
62. wa-mA_dA ta^gidyn fy al-.s.sa.hyfaT .gayr mubArayAt al-rriyyA.daT
63. wa-\'a_hbAr al-`Alam wa-al-brAmi^g al-`ilmiyyaT wa-al-\'iqti.sAdiyyaT
64. wa-al-musalsalAt al-\'i^gtimA`iyyaT
65. ya`ny kul ^say\' mumil ya\'ty \'il_A al-`Alam fy \'A_hir al-llayl
66. wa-\'anta tu.hib kul ^say\' sa.t.hy ya^gy\' \'il_A al-_d_dAkiraT fy \'awwal al-yawm
67. wa-bidAyaT al-nnahAr lina_hru^g min ha_dA al-maw.dU` yA rymA
68. yabdU \'annanA lA nattafiq `alayhi
69. \'anta `al_A .sawAb linatruk ha_dA al-maw.dU`
70. hal baytuki kabyr
71. na`am yata\'allaf min \'arba` .guraf nawm .gurfaT ^gulUs .gurfaT .ta`Am
72. kam .tAbiq fy baytiki yA .sadyqaty
73. yA .sadyqy .tAbiqayn wa-hunAka .hammAm fy kul .tAbiq wa-\'amAma al-bayt .hadyqaT .sa.gyraT
74. laysa hunAka .hadyqaT bi^gAnibi ^siqaty wa-lakinnahA mury.haT wa-na.zyfaT
75. wa-bayty laysa wasi_haN hal ta_dhab li-ziyAraT \'ahlika
76. `indamA tasma.h ly al-fur.saT .gAlibaN natabAdal al-rrasA\'il wa-al-bi.tAqAt
77. _hilAla mawsim al-\'a`yAd wa-fy al-munAsabAt
78. na`am _hA.saTaN `yd al-mylAd wa-`yd al-fi.s.h .hay_tu natabAdal al-hadAyA
79. \'anA \'ufa.d.dil al-ssiyA.haT `al_A ziyAraT al-\'ahl
80. \'u.hibbu \'an \'anAm fy al-funduq
81. \'anA lA \'u.hib al-fanAdiq ka_tyraN hiya .gAliyaT wa-laysat ra_hy.saT

82. wa-lakinnahA ^gamylaT wa-mury.haT

83. \'u.hibbu \'an \'a.s.hU mina al-nnawm fy sA`aT muta\'a_h_hiraT

84. \'a`rif ha_dA \'anA \'a.s_hU bAkiraN kamA qultu laki min qabal wa-\'u.sally .sabA.haN

85. ma` al-\'asaf \'anA lA \'u.hAfi.z `al_A mawA`yd al-.s.salAT wa-lA `al_A maw`id al-.s.siyAm

86. fy al-zzamAn al-qadym kAna al-nnAs ya_dhabUna \'il_A al-^gAmi` wa-al-kanysaT

87. wa-mA zAla al-nnAs fy al-^s^sarq yaf`alUna ha_dA fy ba`.di \'ayyAmi al-\'usbU`

88. wa-lakin laysa kamA fy al-mA.dy faqad \'a.sba.ha al-nnAs yahtammUn bi-al-^s^su.gal kamA fy al-.garb

89. ma`aka .haq al-nnisA\' kAnat ta_dhab lil-zzyAraT \'ammA al-\'An fatu^sAhid al-ttalfiziyUn

90. wa-tastamti` bi-al-\'istimA` \'il_A al-rAdyU wa-al-mUsyq_A wa-al-ssafar wa-al-ssyA.haT

91. \'i_dA kAna al-zzaw^gu .ganiyyaN \'ammA \'i_dA kAna faqyraN

92. fa-al-baqA\' fy al-bayt yakUnu al-qarAr al-nnihA\'iy

93. fy ra\'y ha_dA huwa sabab al-fa^sal al-\'awwal alla_dy yusabbib al-.t.talAq wa-nihAyaT al-`A\'ilaT

94. `an \'ay sabab tatakallam yA ra^gA

95. \'ata.hadda_t `an al-\'ibti`Ad `an al-ll_ah wa-al-ddyn wa-al-_taqAfaT fy kul makAn

96. \'a.zunnu min .hady_tika \'annaka ^sa_h.suN mu.hAfi.z

97. wa-mina al-.garyb \'annany lam \'altaqy bi-fatAT wa-lam \'a_h.tub wa-mA tazawwa^gt

98. kayfa `arifta wa-fahimta mA \'aq.sud \'innaka _daky ^gidaN wa-lakinnaka .gayr wAqi`y

99. wa-lakinnany sa`yd fy .hayAty \'u.hibbu al-masra.h wa-al-ma^sy `al_A ^sA.t_A\'i al-ba.hr

100. hal tata_dakkar kitAba al-kAtibi al-^s^sahyr alla_dy .zahara fy ma^gallaTi \'Ams

101. al-ma^gallaT allaty .sadarat fy al-madynaT \'am fy al-qaryaT

102. lA al-ma^gallaT allaty .tubi`at fy `A.simaT al-wilAyaT

103. wa-allaty tatakallam `an al-`alAqaT wa-al-_hu.tUbaT

104. hal \'anti fi`laN mutazawwi^gaT \'anA fy al-.haqyqaT .gayr mutazawwi^g

105. \'intabih ha_dA huwa al-.gArsUn marraTaN _tAniyaT hal turyd ^say\'aN \'A_har

106. na`am min fa.dlik \'ar.gab fy ba`.d al-^s^sawrabA\' wa-al-ssala.taT wa-al-lla.hmaT wa-al-_hu.dAr

107. yabdU \'annaka ^gU`An \'anA ^gU`AnaT \'ay.daN min fa.dlik mil.h wa-bhAr wa-samak

108. wa-\'uryd ^sawkaT wa-sikkyn wa-mil`aqaT na.zyfaT ha_dihi al-mil`aqaT wasi_haT

109. .hA.dir sayyidaty kayfa tu.hibbyna al-ssamak

110. maqly wa-laysa ma^swy wa-al-_hu.dAr \'uryduhA maslUqaT

111. wa-nuryd al-`a^sA\' bi-sur`aT \'a^s`ur bi-al-^gU`i al-qawy

112. wa-hal tar.gabUna fy ba`.di al-.halw_A ba`da al-wa^gbaT

113. \'i_dA sama.hta ma` al-ka_tyr mina al-ssukkar wa-ba`.d al-.halyb lil-qahwaT ^sukraN

114. \'ar^gU \'an lA \'akUn .talabtu al-ka_tyr mina al-.t.ta`Am

115. lA tahtammy fa-\'anA lastu .ga.dbAnaN \'anA sa`yd li-\'annaki al-\'An zamylaty fy al-\'akl

116. hal tasta\'a^giri al-^s^siqaT allaty taskun fyhA

117. na`am wa\'ahly yusA`idUnany fy daf`ihi fy \'A_hiri al-^s^sahr

118. \'anA \'amluk bayty walakin ya^gib \'an \'adfa` kul sanaT qis.taN min _tamanihi

119. \'i_dan laysa `indaki al-.hurriyaT al-kAmilaT

120. lakinnany \'a^starik ma`a zaw^gy fy daf`i qus.ti al-manzil

121. hal `indaki \'awlAd

122. lA walakin \'anA wa-zaw^gy nanwy _dalika fy al-mustaqbal al-qaryb \'in^sA\' al-ll_ah

123. al-.hamdu li-\'allah \'ana laysa `indy mas\'uwliyyaTa al-\'a.tfAl wa-al-madrasaT wa-al-llibs

124. libsu al-\'a.tfAl \'a.gl_A min libasi al-kibAr wa-madArisuhum .sa`baT

125. ha_dA maw.dU`uN ^sA\'i` ya_dkuruhu mu`.zami al-\'a^s_hA.s fy ^galsAtihim wa-fy al-nnAdy

126. na`am al-\'An \'ahtam bi-^sirA\' al-malAbis ly

127. wa-\'anA \'u.hibbu \'an \'a^stary sayyAraT .sa.gyraT fa-darrA^gaty lam ta`ud munAsibaT

128. hammy al-\'awwal al-\'An huwa al-.hu.sUl `al_A wa.zyfaT fy musta^sf_A

129. mA huwa \'i_hti.sA.suki yA rymA

130. \'anA muta_ha.s.si.saT fy .tubbi al-.t.tawAri_A\' wa-\'u.hibu \'an \'a`mal fy .gurfaTi al-.t.tawAri_A\'

131. ya^gib \'an \'a`Ud \'il_A ^siqqaty al\'An .gadaN huwa `ydu al-mylAd

132. kul `yd wa-\'anta bi-_hayr

133. wa-\'anti bi-_hayr

134. ya^gib \'allA \'ans_A maktaba al-baryd \'ay.daN wa-al-\'i^gtimA` ma`a \'ustA_dy qabla al-`u.tlaT

135. sa\'adfa` al-.hisAb

136. \'abadaN .gayr mumkin \'anA huwa .sA.hib al-dda`waT

137. walakin yA rafyqy \'anA hiya al-.t.tabybaT wa-\'anta huwa al-ttilmy_d

138. wa-\'anA laysa `indy al-mas\'uwliyyaT

139. walakin zaw^gaki wa-rafyqa .taryqi .hayAtiki

140. sAmi.hny yA ra^gA ka_dabtu `alayk

141. hal ha_dA .hulm \'am .haqyqaT wa-\'anti lasti mutazawwi^gaT

142. lA \'aqUlu da\'imaN ha_da li-\'akta^sif \'in kAna al-ra^gul \'a_hlAquhu `AliyaT

143. mA ma`n_A kul ha_dA

144. \'a.zunnu \'anna al-ri^gAla al-yawm yab.ha_tUA `ani al-_d_dahab wa-al-fi.daT wa-al-mAl

145. wa-kayfa wa^gadtiny

146. ^sAbuN mutafawwiq ^gamyl _daky mumti` qawy

147. hal tubAli.gyn

148. `al_A al-\'i.tlAq

149. hal mina al-mumkin \'an naltaqy marraTaN \'u_hr_A fy \'i.hd_A al-\'ayyAm

150. \'in ra.gibta fy _dalika lan takUn ha_dihi al-marraT al-\'a_hyraT allaty naltaqy bihA

151. rubbamA satata`arrafyna `al_A \'ahly wa-\'a.sdiqA\'iy

152. wa-`al_A al-`AdAti al-llubnAniyyaT

153. na`am hiya ^sabyhaT ka_tyraN bi-al-`AdAti al-mi.sriyyaT

154. al-.diyAfaT fy lubnAn maw.dU` ma_tal

155. wa-lA tansy al-karam

The second script is:

1. mar.habaN mA \'ismuka
2. \'ahlaN \'ismI ra^gA wa-\'anti
3. \'anA rImA \'anta .zarIf
4. ^sukraN wa-\'anti ^gamIlaT
5. \'a`rifu ha_dA al-\'amr al-muhim
6. al-ttawA.du` min .sifAtiki \'ay.daN
7. al-.hamIdaTi na`am
8. ta`AlI nazUr al-ma.t`am lina\'kul
9. _haw_haN wa-_hu.draTaN la_dI_daTaN
10. wa-la.hmaN sA_hinaN wa-.halIbaN
11. ha_dihi hadiyyaT _tamInaT
12. al-^s^say\' al-_t_tamIn lil^s^sa_h.s al-.habIb
13. _dahabuN wa-fi.d.daTuN lAmi`aT

14. \'a.zunnu \'annanI ^sa_h.suN .hazIn

15. al-bU\'su wa-al-fiqru min ma`Alim .day`atI

16. lA \'anta ^sa_h.suN .gabI

17. wa-\'anti qalIlaTu al-ttah_dIb

18. yabdU \'annanA lan nattafiq na_hnu al-\'i_tnayn

19. lA ^say\' ya^gma`unA .gayr al-kalAm al-fAri.g

20. wa-al-.t.ta`Am al-lla_dI_d

21. wa-al-.sidfaTu al-.garIbaT

22. wa-.garAbaTu al-\'a.twAr

23. wa-al-kalAm al-fAri.g

24. `ani al-zzawA^g wa-al-.t.talAq

25. \'a`rifu `A\'ilataka

26. min al-.s.sUraT al-latI fI ^gaybI

27. lA min .hadI_tika `anhum

28. hal \'a`^gabUki

29. yabdU \'annahum lu.tafA\'

30. yA zamIlatI mA huwa lawnuki al-mufa.d.dal

31. rubbamA al-\'azraq

32. hal al-.t.ta`miyyaT wa-al-falAfil nafs al-^s^say\'

33. na`am al-_t_tAnI \'ismuN lubnAnI li-.tabaqiN mi.srI

34. fI al-.haqIqaT \'u.hibu \'an \'azUra mi.sr

35. wa-\'anA \'u.hibu \'an \'azUra lubnAn al-`azIz

36. sa\'a_dhab fI al-_t_tAnI min ti^srIn al-\'awwal

37. fI al-`Idi al-_t_tAli_ti wa-al-_talA_tIn

38. hal \'inqa.ta`at `alAqatuka bi-.sadIqatika layl_A

39. \'abadaN sanatazawwa^g .gadaN

40. mabrUk yA .sadIqI

1.2 Creating the Dictionary

The dictionary is basically a file that contains a list of words and their pronunciations. The pronunciations are given in terms of ``phones\'\' and to each phone there will correspond a statistical acoustic model. Before the USMA wrote the dictionary, they had to decide on a list of

phones that we wanted to model. Another Arabic instructor at the Academy, LTC. Terrence Potter, produced a list of the most important phones in the Arabic language. We classified the phones by their articulatory features, because we use these features later in the process of clustering triphones.

Arabic Phone	Articulatory Features
A	stressed mid central vowel
AA	stressed low front vowel
C	voiced pharyngeal fricative
D	velarized voiced alveolar stop
G	voiced velar fricative
H	voiceless pharyngeal fricative
I	stressed high front lax vowel
II	stressed high front tense vowel
Q	voiceless glottal stop
S	velarized voiceless alveolar fricative
T	velarized voiceless alveolar stop
TH	velarized voiced interdental fricative
U	stressed high back rounded lax vowel
UU	stressed high back rounded tense vowel
Z	voiced interdental fricative
a	unstressed mid central vowel
aa	unstressed low front vowel
b	bilabial voiced stop
d	voiced alveolar stop
dj	voiced alveolar affricate
e	upper mid front tense vowel
f	voiceless labiodental fricative
g	voiced velar stop
h	voiceless glottal fricative
i	unstressed high front lax vowel
ii	unstressed high front tense vowel

Arabic Phone	Articulatory Features
j	voiced palato-alveolar fricative
k	voiceless velar stop
l	voiced alveolar lateral
m	voiced bilabial nasal
n	voiced alveolar nasal
q	voiceless uvular stop
r	voiced alveolar flap
s	voiceless alveolar fricative
sh	voiceless palato-alveolar fricative
sil	silence
sp	short pause
t	voiceless alveolar stop
th	voiceless interdental fricative
u	unstressed high back rounded lax vowel
uu	unstressed high back rounded tense vowel
w	voiced bilabial approximant
x	voiceless velar fricative
y	voiced palatal approximant
z	voiced alveolar fricative

Table 3 - Arabic Phones and Features

1.3 Recording the Data

USMA collected the speech data at four different sites. Native Arabic speakers who were learning English as a second language at the San Antonio branch of the Defense Language Institute donated their speech to the native corpus. Mr. Chouairy headed a team that collected more data from members of a native Arabic speaking community near Toronto Canada. Army linguists from Fort Bragg and the Marshall Center in Garmisch Germany contributed their speech to the corpus of nonnative speech. Pentium processor speed laptop computers running Windows NT were used with a brand name microphone. The sampling rate was set at 22050 Hz and 16-bit audio was used. The WinCalis script presented a line of text in the Arabic script and then it played a .wav file of Mr. Chouairy's rendition of the sentence. When ready, the informant pressed the enter key and read the prompt. WinCalis played the recording so that the informant could either move on to the next prompt if the recording was good, or re-record the prompt in

case of a bad recording. Some of the native informants read every sentence from the WinCalis script, but most read either the first 90 sentences or the last 90 sentences. The nonnative informants at least attempted to read all 40 of their prompts. USMA ended up with approximately 5300 .wav files from the native informants and approximately 1200 files from the nonnative informants. Of the native informants 18 were women and 40 were men. Of the 22 nonnative informants 8 were females and 14 were males.

1.4 Creating the Transcription Files

In order to compute statistics for the acoustic models, each recording or data file must have a corresponding transcription file. These transcriptions are called label files and they come in many formats. They can be sentence level, word-level, syllable-level, or phone-level transcriptions. They can be time aligned or not. In our case the goal was to create simple phone-level label files with no time alignment. USMA created the phone-level label files in two steps. First a perl script was used to convert the file containing the prompts into a set of word-level files. The file with the prompts was written with one prompt on each line. The perl script looked at one line at a time, creating a new file for the line and placing each word in the prompt on a separate line in the new file. So in the case of the native prompt script there were 155 files each containing one word on each line. Then the HTK label editor tool HLEd was used to create phone level files. Loosely speaking HLEd uses the dictionary to ``look up" the pronunciation of each word in the word level files, then it replaces the word with the word's pronunciation and each phone gets placed on its own line. As an example, we show here the first sentence of the native prompt script and its phone level label file:

```
mar.habaN mA \'ismuka

sil

m mar.habaN

A

r

H

a

b

a

n

sp

m mA

AA

sp

Q "'ismuka"

I
```

s

m

u

k

a

sil

.

1.5 Coding the Data

In speaking, a person encodes ideas, represented by symbols, into acoustic forms that are called phones. In listening, a person recovers the symbols that represent ideas from the acoustic data, HTK uses digital signal processing (dsp) algorithms to model this encoding--decoding process. USMA followed the HTKBook recommendation of encoding the data with Mel Frequency Cepstral Coefficients (MFCC) with energy delta-energy and acceleration parameters.

Step 2.0 Creating Monophone HMMs

Again USMA followed very closely the HTKBook's recommendations in designing the Hidden Markov Models (HMMs). Models with 5-states (only three emitting) and single mixture gaussian distributions were used.

2.1 Creating Flat Start Monophones

Since USMA was modeling phones and since the speech data consists of spoken sentences which can be very long sequences of phones, a major task for the algorithms that calculate the statistics for the models was to find the endpoints in time of the data for a given model. One way to do this is to use a program that enables a person to view and hear the speech and to mark by hand the endpoints of the phones. USMA did not accumulate enough of this hand-labeled data; so they decided to follow the flat start training strategy. In this strategy, statistics are calculated over all of the data and these statistics are assigned to all of the models. After this step, all of the models have the same statistical parameters. Then subsequent algorithms attempt to automatically time align the data.

2.2 Fixing the Silence Models

Small abnormalities usually creep into speech. For example, persons sometimes make false starts at the beginning of words or make unintended sounds. To make the speech recognizer tolerant of these abnormalities, the HTKBook introduces a short pause model and a modification to the silence model. To train the short pause (sp) model USMA had to go back and insert the sp symbol in the label files.

Step 3.0 Creating Tied State Triphones

The context of a phone, i.e. the phones that precede and follow it, influence its acoustic realization. The goal is then to model phones in their contexts. Triphone models are an attempt to achieve this goal. The HTK training tools require triphone level transcriptions. One of the drawbacks of modeling triphones is the large number of triphones that can be formed from a small number of monophones. In this case, there were 44 monophones and 44^3 = 85184 triphones. It was not reasonable to expect to ever have enough training data to properly train 85184 individual triphones. The idea then was to train very similar triphones on the same data. But a question arose regarding how to decide which triphones are similar. HTK provides a method for clustering similar triphones into groups and for tying these groups to the same training data. The clustering method uses the phonetic articulatory features given in the table above to classify the triphones.

Step 4.0 Creating the Language Model

Speech recognition systems need more than just an acoustic model and a dictionary in order to produce text from speech. They also require a language model. USMA wrote the Arabic language model from the English specifications for the microworld.

Step 5.0 Recognizer Evaluation

To test the Arabic recognizer, USMA used 875 recordings from 21 nonnative speakers. Each informant read from a list of 40 sentences. The standard sampling rate of 22050 Hz was used. The testing was performed on a Pentium P2, 300MHz computer with 128 MB RAM, running the linux operating system. Three different lattices were used for testing.

1. A lattice containing a list of the 40 test sentences.
2. A much larger lattice into which we embedded the 40 test sentences.
3. A word loop containing the 126 words in the script.

The tests were performed on models with 9 mixture components. The percentage of correct sentences recognized increased as we increased the number of mixture components. The delay between an utterance and recognition seems to increase as the mixture components increase, but USMA has not measured this yet. Performance also seems to depend on the architecture.

Lattice	**Perplexity**	**% Of Correct Sentences**	**% of correct words**
40 sentences	2.826634	99.66	99.67
Microworld + 40 sentences	9.670476	92.46	91.76
Wordloop	108.095966	45.14	83.69

Task 8: Develop Arabic, Spanish and English CSR Components

For continuous speech recognition using HTK, each language requires the following components:

- A set of acoustic models (HMMs)
- A dictionary mapping the required words into the acoustic models
- A network specifying the recognition grammar
- A configuration file
- An application program

Acoustic Models

The Arabic acoustic model development is described in Task 7. The English and Spanish acoustic models were developed by Entropic Cambridge Research Laboratory (ECRL) and are supplied with the speech recognition development tools.

The English phone set (Table 4) consists of 41 distinct speech phones plus the silence and short pause phones.

Symbol	**Example**	**Symbol**	**Example**
aa	balm	b	bet
aa	box	d	debt
ah	but	k	cat
ao	bought	p	pet
aw	bout	t	tat
ax	about	dh	that
ay	bite	th	thin
eh	bet	f	fan
er	bird	v	van
ey	bait	s	sue
ih	bit	sh	shoe
iy	beet	z	zoo
ow	boat	zh	measure
oy	boy	ch	cheap
uh	book	jh	jeep
uw	boot	m	met
l	led	n	net

Symbol	Example	Symbol	Example
r	red	en	button
w	wed	ng	thing
y	yet	sil	silence
hh	hat	sp	short pause

Table 4 - English Phone Set (Power, 1997)

The Spanish phone set (Table 5) contains 25 distinct speech phones plus the two silence phones.

Symbol	Example	Symbol	Example
a	casa	b	boca
a	está	d	dolor
e	dijé	g	gota
i	vino	k	cama
o	habló	p	peso
u	sucre	t	techo
w	güera	f	fino
y	ayer	j	gel
l	lago	s	asa
ll	llave	z	azote
r	aro	m	mano
rr	arroz	n	nombre
sil	silence	sp	short pause

Table 5 - Spanish Phone Set (Power, 1997)

<u>Dictionaries</u>

The Arabic dictionary development is described in Task 7. The English and Spanish dictionaries used in MILT are subsets of the dictionaries developed by Entropic Cambridge Research Laboratory supplied with the speech recognition development tools.

The ECRL supplied English and Spanish dictionaries consist of 90,000 words. MA&D used the full dictionaries while developing the CSR components and built smaller dictionaries for use in the MILT application. The smaller dictionaries contain only the words used in the

microworld exercise. The English dictionary has 290 words, the Spanish has 214 words, and the Arabic has 495 words.

Networks

Language models for all languages were developed by MA&D using the Entropic HTK grapHvite software. grapHvite is a developer kit for building small to medium vocabulary speech recognition systems. It provides the means for creating the recognizer components for a given speech recognition task.

In particular, the Netbuilder portion of graphvite was used. Netbuilder is a visual tool for rapidly creating a syntax network and dictionary for a task-specific recognizer. The Netbuilder provides tools for testing that a network, once created, is syntactically correct and also for interactively testing a network with the grapHvite decoder. In addition, the Netbuilder allows pronunciations to be edited or pronunciations for words not covered by any of the supplied dictionaries.

Configuration Files

Configuration files control various parameters of the speech recognizer. Each language required a unique configuration file to be developed. The configuration file sets general parameters, directory defaults, suffix defaults and system configurations.

Application Program

The MILT microworld contains application code written in C++ to initialize the speech recognition and process results. In addition, Java application programs were developed to test the speech recognition components separate from the microworld.

Task 9: Expand the Arabic NLP System

The Natural Language Processor (NLP) embedded in MILT was developed by the University of Maryland. The NLP works by parsing a sentence in a bottom-up process by identifying the individual words and then determining the relationships among them. The main components of the parser are a preprocessor, a morphological analyzer, a lexicon, a syntactic parser, an error handling facility, and a semantic interpreter. First, word strings are submitted to an interactive preprocessor that identifies spelling mistakes. The morphological analyzer then decomposes the words into subparts (i.e., roots and affixes) based on information in the lexicon (similar to a dictionary). Specific information is drawn from the lexicon about the word subparts (e.g., whether it is a verb or noun, singular or plural). The word subparts are unified back into their original state and passed along with the descriptive information from the lexicon to the syntactic parser.

Based on this information, the parser tries to build a structure, called a parse tree, which reflects the appropriate relationship among the words. The parser moves words around trying to find all possible constructions that satisfy a minimal set of basic phrase structure rules (i.e., language universal abstract principles). Also operating are two interacting modules. The required constraints component establishes those barest essentials needed to comprise somewhat intelligible parts of a sentence. The second component, the preferred constraints, contains the categories of grammatical errors to flag once the sentence or phrase has been successfully parsed.

In other words, the parser at first overgenerates the number of possible structures and then imposes constraints ("weeds them out") based on language specific rules. This approach provides for robustness (i.e., parser does not fail as it encounters each error) and allows specification of the grammatical error types of importance to the instructional objectives.

Let us illustrate using an example in English, "The girl write the story." All the words and subparts are recognized and passed on to the parser. "Write" has been identified as a verb that often has an inanimate object. The parser looks for an inanimate object in the appropriate position (following the verb) to complete the verb phrase branch of the tree, "write story". It continues on in this manner until it completes the parse tree and the error handler conveys to the tutor that a subject-verb agreement error ("girl-write") was made. The tutor reads the error data and uses that information to drive specific feedback to the student and to build the student model.

The parser can even handle some reasonable sentence fragments, so that if the tutor asked where the castle is located, the student could appropriately reply, "to the north of Berlin". However, every possible sentence construction and grammatical error could not be captured by the NLP. Therefore, determination of the range of input and error types was made through analysis of language usage by the military linguists in job-specific environments. The most common, problematic, and critical areas identified were those which were included in the NLP development

The Natural Language Processor Subsystem can be broken down into the following parts:

- Lexicon
- Parser
- Lexical Conceptual Structure (LCS)
- Semantic component

Communication within the NL system is as follows: input is passed to the lexicon, lexicon items go to the parser, the parser produces a parse structure which is passed to the LCS system, and the LCS system passes information to the semantic component for matching or for discourse. Almost all communication is "point to point," that is, components directly call or return values to other subsystems. The one exception is the communication of information from the parser to the LCS, which must go through a special C function call, as there is no direct communication from Prolog to Lisp.

This task involved expanding the Arabic NLP system used in MILT version 1.0 to support the new objects and actions in the 3D microworld.

The new entries are:

Surface	Root	Collocations	Gloss	Category	Primitive
>Hml	Hml<ao-N Hml<io-N	'ilaY Null	carry	V	cause go_loc
<SEd	SEd<Iia	EalY Ean fwq<ao-N Null	climb	V	go_loc
<nzl	nzl<Iia nzl<Iai	Ean taHt	Descend	V	go_loc
>dxl	dxl<IV dxl<Iau	min fI Null	Enter	V	go_loc
>glq	glq<Iai glq<IV	Null	Close	V+ed	cause go_ident
gTy	gTw<Iau gTw<II	bi	Cover	V+ed	cause go_ident
zHf	zHf<ao-N	fI min taHt	Crawl	V	go_loc
<$rb	$rb<Iia	Null	Drink	V	cause go_ident
<rm	rmX<Iai	Null	Throw	V	cause go_loc
>kl	'kl<Iau 'kl<II	Null	Eat	V+ed	cause go_ident
qf	wqf<Iai	Ean	Stand	V	go_loc
<jlb	jlb<Iai	min Null	Bring	V	cause go_loc
q@mp	qwm<aaap- Nap	Null	inventory	N	
>ET	ETy<IV	li	Give	V	cause go_loc
>rjE	rjE<Iai	Null	return	V	go_loc
<*hb	*hb<Iaa	'ilaY min fI	go	V	go_loc
$rq	$rq<ao-N	Null	east	N	
$m@l	$ml<aA-N	Null	north	N	
ys@r	ysr<aA-N	Null	left	N	
jnwb	jnb<aU-N	Null	south	N	
grb	grb<ao-N	Null	West	N	

Surface	Root	Collocations	Gloss	Category	Primitive
ymyn	ymn<aI-N	Null	Right	N	
Elq	Elq<Iia Elq<V	EalY	Hang	V+ed	cause go_ident
Swrp	SWr<uoap-Napdu	Null	Picture	N	
Hy@T	HwT<iA-N HwT<=ap1-Ndu	Null	Wall	N	
smE	smE<Iia smE<II	Null	hear	V	go_perc
>Drb	Drb<Iai	Null	hit	V+ed	cause go_ident
<msk	msk<Iai	Null	Hold	V	cause stay_loc
>dxl	dxl<IV	fI Null	insert	V	cause go_loc
qfz	qfz<Iai	fwq<ao-N Ean EalY Null 'ilaY	Jump	V	go_loc
>lbT	lbT<Iai	Null	Kick	V+ed	cause go_ident
qrE	qrE<Iaa	Null EalY	Knock	V	go_loc
b@b	bwb<aa-Ndu	Null	Door	N	
T@wlp	TAWil	Null	Table	N	
@vb	wvb<Iai	fwq<ao-N EalY	Leap	V	go_loc
>trk	trk<Iau	EalY Null	Leave	V	go_loc
<rfE	rfE<Iaa	Null	Lift	V	go_loc
<stmE	smE<VIII	'ilaY	Listen	V	go_perc
jhz	jhz<II	Null	Load	V	cause go_loc
$ryT	$rT<aI-Ndu	Null	Cassette	N	
>qfl	qfl<Iai qfl<IV	Null	Lock	V+ed	cause go_ident

Surface	Root	Collocations	Gloss	Category	Primitive
>nZr	nZr<Iau	Null 'ilaY wry<aA-n taHt fI fwq<ao-N	Look	V	go_perc
wr@'	wry<aA-N	Null	behind	P	

Table 6 - New Arabic NLP Entries

Task 10: Deliver the software

Interim versions of the software were delivered to ARI throughout the contract. The final software delivery date was January 31, 1999.

Task 11: Develop an Arabic continuous speech recognition exercise

A sample microworld exercise was developed for this task. In the sample exercise, students search a room of an enemy prisoner. Student commands result in movement of objects around the room. The goal of the exercise is for the student to find a letter that reveals the location of the city to which the enemy forces intend to move.

Directions: Our intelligence reports tell us that there is to be an attack launched soon and that information about this attack is located in this room. You are to search this room to find out where the attack is going to occur. You may have to look in objects (cabinets, boxes, etc.) to find the documents describing the attack.

Question: Where are the troops moving?

Answer: Srqnd

List of Objects:

file cabinet - visible

table - visible

waste basket - visible

box - visible

briefcase - visible on floor

lamp - visible on table

radio - visible on table

envelope - in briefcase

letter - in envelope

gun - in file cabinet

map - in box

newspaper - in file cabinet

book - in briefcase

Radio: When student turns on the radio, he will hear the following :

"Good Morning Halabja! The following song is dedicated to the brave men who are fighting for our just cause near the great river. We pray God that you are safe and strong."

Newspaper:

headline reads:

Troop Morale is High

text:

The soldiers are enjoying a sense of great confidence because of their excellent training in the use of modern weapons and also because they have trust in our Air Force and its capability to clear our skies of ennui planes and provide air coverage for our ground forces.

Book:

Title:

Infantry Code Manual

text page 1:

January - February 1999

Sensitive Classified Information

text page 2:

frequency	call sign
100.3	CEOI
97.5	XR29
85.0	J2Q

Envelope:

Leila Sadawi

Arab Republic

Najran

1441 al-Hadeekah St

Letter:

My Beloved Leila:

I am writing to you from my position on the border. I am very well. Everything here is in a state of readiness for war. When I got here I helped dig trenches and carrying food-supplies. Then we resumed our daily training. Every now and then, I go on reconnaissance missions in Halabja.

I miss you so much and want this war to be over so I can see you. I do not know when this will happen, but it is likely that the war will be a long one because we are about to move towards Srqand. This will make the war range on a larger scale and we may also make more movements.

I don't want you to worry too much about me. My hope is so great in being able to see you again after the war is over. Look after yourself and do not forget me or forget writing to me. It is unfortunate that your letters reach me late, but do not keep me bereaved of your sweet words.

Love,

Saaber

Task 12: Prepare system documentation and user help

Documentation prepared for MILT 2.0 under this task include the "MILT 2.0 Software User's Manual," the author help file, the student help file, the main help file, the 3D microworld authoring help file, a author tutorial, grammar references for Arabic and Spanish, and the final report.

Task 13: Complete monthly progress reports

MA&D provided monthly status reports during the project on the 15th of each month. The reports each contained the following information:

- Progress during the month including a summary of completed and on-going activities
- Progress projected for the following month
- Problems encountered or anticipated
- Costs, both direct and indirect incurred

Task 14: Complete final report

This document is the result of this task.

Summary

This report documented the design and development of an authorable, speech recognition enhanced, three-dimensional microworld. The speech recognition incorporated into MILT is corpus-based, continuous and speaker-independent. Major components of the developed system are continuous speech recognition components for English, Arabic, and Spanish, an authorable 3D microworld, and an expanded Arabic natural language processing (NLP) system.

References

Odell, J., Ollason, D., Valtchev, V., Whitehouse, D. "The HAPI Book: A description of the HTK Application Programming Interface,Version 1.2" Entropic Cambridge Research Laboratory, Ltd., December 1997.

Ollason, D., Whitehouse, D., Morton, R. "The IHAPI Book: A description of the ISLE Application Programming Interface, Version 1.1" Entropic Cambridge Research Laboratory, Ltd., July 1997.

Plott, B., "Software User's Manual for the Military Language Tutor (MILT) Windows - Based Tutoring System," Contract Delivery to U.S. Army Research Institute, Contract No. MDA903-92-C-0229, Delivery Order No. 0007, Revision D: August 1997.

Plott, B., Princen, E. "MILT 2.0 Design Document," Contract Delivery to U.S. Army Research Institute, Contract No. DASW01-97-C-0039, June 1997.

Power, K., Morton, R., Matheson, C., Ollason, D., "The grapHvite Book, Version 1.1" Entropic Cambridge Research Laboratory, Ltd., April 1997.

Open Inventor for Win32 On-Line Help Documentation. Template Graphics Software. Copyright 1997.

Appendix A

ArabTeX Conventions

ArabTeX is a package extending the capabilities of TeX/LaTeX to generate the arabic writing from an ASCII transliteration for texts in several languages using the arabic script.

Standard arabic and persian characters:

----

b bah

d dal

.s ssad

f fah

h hah

' hamza

t tah

_d dhal

.d ddad

q qaf

w waw

N tanween

_t thah

r rah

.t ttah

k kaf

y yah

Y alif maqsoura

^g geem

z zay

.z tthah

l lam

g gaf

_A alif maqsoura

.h hhah

s seen

` `ain

m meem

p pah

T tah marbouta

_h khah

^s sheen

.g ghain

n noon

v vah

W waw (see below)

Additional characters generally available:

----

c hhah with hamza

^c gim with three dots (below)

,c khah with three dots (above)

^z zay with three dots (above)

~n kaf with three dots (Ottoman)

~l law with a bow accent (Kurdish)

~r rah with two bows (Kurdish)